



**VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ**

BRNO UNIVERSITY OF TECHNOLOGY

**FAKULTA PODNIKATELSKÁ**

FACULTY OF BUSINESS AND MANAGEMENT

**ÚSTAV INFORMATIKY**

INSTITUTE OF INFORMATICS

**NÁVRH DÍLČÍ ČÁSTI INFORMAČNÍHO SYSTÉMU**

PROPOSAL OF PART OF INFORMATION SYSTEM

**DIPLOMOVÁ PRÁCE**

MASTER'S THESIS

**AUTOR PRÁCE**

AUTHOR

**Bc. Jiří Bašta**

**VEDOUCÍ PRÁCE**

SUPERVISOR

**Ing. Jan Luhan, Ph.D., MSc**

**BRNO 2021**

# Zadání diplomové práce

Ústav: Ústav informatiky  
Student: **Bc. Jiří Bašta**  
Studijní program: Systémové inženýrství a informatika  
Studijní obor: Informační management  
Vedoucí práce: **Ing. Jan Luhan, Ph.D., MSc**  
Akademický rok: 2020/21

Ředitel ústavu Vám v souladu se zákonem č. 111/1998 Sb., o vysokých školách ve znění pozdějších předpisů a se Studijním a zkušebním řádem VUT v Brně zadává diplomovou práci s názvem:

## Návrh dílčí části informačního systému

### Charakteristika problematiky úkolu:

Úvod  
Cíle práce, metody a postupy zpracování  
Teoretická východiska práce  
Analýza současného stavu  
Vlastní návrhy řešení  
Závěr  
Seznam použité literatury  
Přílohy

### Cíle, kterých má být dosaženo:

Navrhnout a vyvinout část informačního systému v podobě webového obchodu pro konkrétní subjekt s využitím nové vývojové strategie.

### Základní literární prameny:

BRUCKNER, T. Tvorba informačních systémů: principy, metodiky, architektury. 1. vyd. Praha: Grada Publishing, 2012. 360 s. ISBN 978-80-247-4153-6.

LAURENČÍK, M. Tvorba www stránek v HTML a CSS. 1. vyd. Praha: Grada Publishing, 2019. 224 s. ISBN 978-80-271-2241-7.

SODOMKA, P. a H. KLČOVÁ. Informační systémy v podnikové praxi. 2. aktual. a rozš. vyd. Brno: Computer Press, 2011. 504 s. ISBN 978-80-251-2878-7.

ŠOCHOVÁ, Z. a E. KUNCE. Agilní metody řízení projektů. 1. vyd. Brno: Computer Press, 2014. 176 s. ISBN 978-80-251-4194-6.

WELLING, L. a L. THOMSON. Mistrovství PHP a MySQL. 1. vyd. Brno: Computer Press, 2017. 800 s. ISBN 978-80-251-4892-1.

Termín odevzdání diplomové práce je stanoven časovým plánem akademického roku 2020/21

V Brně dne 28.2.2021

L. S.

---

Mgr. Veronika Novotná, Ph.D.  
ředitel

---

doc. Ing. Vojtěch Bartoš, Ph.D.  
děkan

## Abstrakt

Tato práce se zabývá návrhem a implementací dílčí části informačního systému ve spolupráci s integrováním nového agilního projektového řízení. Dílčí částí informačního systému je webový obchod pro společnost, poskytující IPTV. Tento webový obchod bude implementován v jazyce PHP za využití frameworku Nette. Nové projektové řízení, jež je průběžně integrování je agilní metodika SCRUM.

## Abstract

This thesis is focused on design and implementation of partial part of the information system in cooperation with the integration of a new agile project management. Partial part of the information system is a web shop for company providing IPTV. This web shop will be implemented in PHP language and Nette framework. The new project management, which will be continuously integrated, is agile SCRUM methodology.

## Klíčová slova

Informační systém, IPTV, projektové řízení, agile, scrum, PHP, Nette, webový obchod, e-shop

## Keywords

Information system, IPTV, project management, agile, scrum, PHP, Nette, web shop, e-shop.

## Citace

BASTA, Jiří. *Návrh dílčí části informačního systému*. Brno, 2021. Diplomová práce. Vysoké učení technické v Brně, Fakulta podnikatelská. Vedoucí práce Ing. Jan Luhan, Ph.D., MSc

# Návrh dílčí části informačního systému

## Prohlášení

Prohlašuji, že jsem tuto diplomovou práci vypracoval samostatně. Další informace mi poskytla společnost ve které pracuji SledovaniTV s.r.o.. Prohlašuji, že jsem uvedl všechny literární prameny, publikace a další zdroje, ze kterých jsem čerpal.

.....

Jiří Basta

15. května 2021

## Poděkování

Rád bych zde poděkoval svému vedoucímu panu Ing. Janu Luhanovi, PhD., MSc. za jeho pomoc a poskytování rad a zpětné vazby při vypracování této práce.

# Obsah

<b>Úvod</b>	<b>10</b>
<b>1 Vymezení problému a cíle práce</b>	<b>11</b>
<b>2 Teoretická východiska práce</b>	<b>12</b>
2.1 Data a informace . . . . .	12
2.1.1 Data . . . . .	12
2.1.2 Informace . . . . .	13
2.2 Informační systémy . . . . .	13
2.2.1 Podnikové informační systémy . . . . .	14
2.2.2 Veřejné informační systémy . . . . .	14
2.3 Technologie . . . . .	14
2.3.1 HyperText Markup Language (HTML) . . . . .	14
2.3.2 Cascading Style Sheets (CSS) . . . . .	15
2.3.3 JavaScript . . . . .	15
2.3.4 PHP . . . . .	16
2.3.5 Relační databáze . . . . .	19
2.3.6 Apache . . . . .	20
2.3.7 Git . . . . .	20
2.3.8 IPTV . . . . .	21
2.4 Projektové řízení . . . . .	21
2.4.1 Tradiční metodika . . . . .	22
2.4.2 Agilní metodika . . . . .	22
<b>3 Analýza problému a současné situace</b>	<b>26</b>
3.1 Představení společnosti . . . . .	26

3.1.1	Struktura společnost . . . . .	28
3.1.2	Partneři a provozovatelé . . . . .	29
3.1.3	Účetnictví . . . . .	29
3.2	Informační systém ve společnosti . . . . .	30
3.2.1	Uživatelská sekce . . . . .	30
3.2.2	Administrace systému . . . . .	32
3.2.3	Partnerská administrace . . . . .	32
3.2.4	Přizpůsobení . . . . .	33
3.2.5	Statické stránky . . . . .	34
3.2.6	Služby . . . . .	35
3.2.7	Internetový obchod . . . . .	37
3.3	Analýzy firmy . . . . .	37
3.3.1	SLEPTE analýza . . . . .	37
3.3.2	Porterova analýza pěti sil . . . . .	41
3.3.3	Analýza interních faktorů pomocí modelu 7S . . . . .	43
3.3.4	SWOT analýza . . . . .	47
3.3.5	Výstupy analýz . . . . .	50
<b>4</b>	<b>Návrh řešení a jeho přínos</b>	<b>51</b>
4.1	Aktuální stav . . . . .	51
4.2	Požadavky . . . . .	52
4.3	Lewinův model řízení změn . . . . .	53
4.3.1	Síly působící na změnu . . . . .	53
4.3.2	Agent změny . . . . .	54
4.3.3	Intervenční oblasti . . . . .	55
4.3.4	Fáze rozmrazení . . . . .	56
4.3.5	Fáze provádění změn . . . . .	56
4.3.6	Fáze zmrazení . . . . .	56
4.4	Časová analýza . . . . .	56
4.4.1	Časové milníky . . . . .	57
4.4.2	Časový graf . . . . .	57
4.5	Nákladová analýza . . . . .	59
4.5.1	Školení agile . . . . .	59

4.5.2	Technologie a nástroje . . . . .	59
4.5.3	Prostory . . . . .	59
4.5.4	Vývoj . . . . .	60
4.5.5	Souhrn nákladů . . . . .	60
4.6	Návrh a zavedení projektového řízení . . . . .	61
4.6.1	Stand-up . . . . .	62
4.6.2	Ceremonie . . . . .	63
4.6.3	Sprint . . . . .	64
4.6.4	Nástroj Jira . . . . .	65
4.6.5	GitLab . . . . .	67
4.7	Návrh a zavedení webového obchodu . . . . .	72
4.7.1	Návrhová část . . . . .	72
4.7.2	Realizace . . . . .	79
4.8	Přínosy . . . . .	92
	<b>Závěr</b>	<b>95</b>
	<b>Literatura</b>	<b>97</b>



# Úvod

Tato diplomová práce se zaměřuje především na návrh a implementaci dílčí části informačního systému, konkrétně internetového obchodu pro společnost, jež poskytuje televizní vysílání pomocí internetového připojení tzv. IPTV (viz kapitola 2.3.8). Dále je součástí této práce popis současného stavu projektového řízení v dané společnosti, různé úskalí a následné zlepšení tohoto řízení pomocí agilních vývojových metod (SCRUM). Nové projektové řízení bude ve společnosti aplikováno zároveň s vývojem této dílčí části systému. První část práce je věnována obecné teorii k dané tématice, informačním systémům, programovacím jazykům a metodám použitých pro vývoj této části systému. Dále jsou tu popsány nástroje pro vývoj, organizaci práce a projektové řízení.

Následně je popsána zvolená společnost, její **současný stav a struktura řízení**. Pro správné pochopení problematiky zde bude popsána struktura firmy relativně detailně. Mimo struktury se zde bude nacházet řada **analýz pro vyhodnocení aktuálního stavu společnosti**. Analýzy budou zohledňovat různé pohledy na společnost. Analyzováno bude například firemní prostředí a to vnitřní i vnější a také celková konkurenceschopnost společnosti.

Finální kapitola se zabývá samotným návrhem řešení a jeho přínosem. Tato kapitola je rozdělena do dvou částí. Návrhu a integrace nového **projektového řízení** a návrh a implementace **elektronického obchodu**. Jsou zde vypracovány také některé důležité analýzy jako časová a nákladová analýza. Nakonec jsou popsány přínosy, kterých bylo dosaženo touto inovací webového obchodu a zavedení nového projektového řízení.

# Kapitola 1

## Vymezení problému a cíle práce

- Návrh a implementace dílčí části informačního systému - elektronického obchodu
- Popis a návrh nového projektového řízení
- Integrace agilní projektového řízení - SCRUM

Hlavním cílem této práce je návrh dílčí části informačního systému a v tomto konkrétním případě to bude návrh a implementace internetového obchodu pro společnost poskytující IPTV (viz kapitola 2.3.8). Tento internetový obchod se zabývá především poskytováním služeb seskupených do tzv. balíčků. Tyto balíčky nabízí klientům možnost sledovat živé televizní vysílání za různé flexibilní ceny. Mimo poskytování televizního vysílání, jsou v těchto balíčcích také další výhody - jako například zpětné sledování (archív), databáze filmů, nahrávání pořadů či využívání různých prémiových služeb, jako HBO, kanály pro dospělé apod. Lze říci, že se jedná spíše o modernizaci tohoto elektronického obchodu. Ale z důvodu takto rozsáhlé modernizace a velkého počtu změn je tedy vyvíjen tento obchod jako nová separovaná část. Jedním z důležitých bodů této inovace je především široká přizpůsobivost jednotlivých položek a balíčků v obchodě.

Z důvodu takto rozsáhlé změny informačního systému je vyžadovaná vysoká úroveň projektového řízení. Bez efektivního projektového řízení by návrh a implementace této změny mohlo zabrat několikanásobně více času, či případně nemuselo být vůbec realizovatelné. Druhým cílem je tedy popis současného projektového řízení ve společnosti a postupné integraci agilního vývoje (SCRUM). Jelikož je projektové řízení důležitou částí vývoje a ve větších společnostech se bez správně zavedeného projektového řízení nedá příliš efektivně spolupracovat, tak je tohle téma velice úzce spjaté i se samotným návrhem a vývojem dílčí části informačního systému - internetovým obchodem.

## Kapitola 2

# Teoretická východiska práce

Zde v této sekci budou vysvětleny základní pojmy všech použitých technologií, návrhových modelů či různých metod o kterých tato práce pojednává. Konkrétně zde bude několik hlavních částí, jakožto obecný popis informačního systému, použité technologie, návrhový vzor a popis projektového řízení.

### 2.1 Data a informace

Pro správné vysvětlení následující kapitoly je důležité správně definovat pojmy data a informace. Tyto pojmy jsou následně často využívány v kapitole 2.2.

#### 2.1.1 Data

Data, respektive podniková data, představují neodmyslitelný prvek podnikového informačního systému. Jsou nositeli zaznamenaných skutečností souvisejících s aktivitami podniku a zároveň jsou schopna přenosu, interpretace a zpracování [16]. Data lze chápat také v obecnějším měřítku kdy se jedná o údaje, které jsou následně různě zpracovávány. Tyto údaje mohou sloužit pro popis nějakého jevu nebo vlastnosti objektu.

Data lze rozdělit do tří základních skupin:

- **data o společenských podmínkách podnikání** - zahrnují veškeré poznatky o mikro a makrookolí organizace, jako jsou zaznamenané údaje o demografických, sociálních, ekonomických trendech společnosti, pracovní síle, dostupnosti materiálu, kapitálu a dalších faktorech ovlivňujících hodnototvorný řetězec firmy [16]

- **data o trhu** - skládají se o evidovaných skutečnostech o nabídce, poptávce, konkurenci a celkovém dění na trhu [16]
- **interní data** - obchodní a finanční plány různé predikce vývoje, data o podnikových zdrojích a jejich alokaci a omezení [16]

### 2.1.2 Informace

I zde se lze opět dívat na pojem informace z několika úhlů. Prvním by byl obecnější pohled, to znamená, že pojem informace lze vysvětlit jako interpretaci dat v určitém kontextu, aby dávala smysl pro příjemce. Příjemce zde může být ať už osoba nebo klidně i jiná technologie (systém). Informace jsou tedy zpracovaná data.

Z pohledu podnikových informačních systémů se tento pojem příliš neliší. Je pouze zaveden jiný kontext, do kterého je tento pojem zasazen. K samotným datům je nutné přidat ještě znalost. Pro někoho mohou data tvořit informaci, ale pro jinou stranu se může jednat o prázdná data, která nedávají smysl. Pro správnou interpretaci informací je tedy nutná **znalost**. Tato znalost je relativní a pro každého příjemce může být jiná. Řízení firem a organizací vyžaduje kvalitní informace. Jedná se o klíčové vstup a výstupy každé organizace. Například pokud nemám všechny dostupné informace o lidech, financích, materiálu, infrastruktuře nebo o konkurenci, nelze organizaci správně řídit. Správně interpretované informace jsou tedy velice důležité pro fungování [16].

## 2.2 Informační systémy

Tato práce je z velké části věnována návrhu a implementaci dílčí části informačního systému. Proto je důležité popsat co vlastně informační systém (IS) je a k čemu je dobrý.

Informační systém lze chápat jako systém vzájemně propojených informací a procesů, které s těmito informacemi pracují [5]. V tomto kontextu **procesy** znamenají funkce, které zpracovávají samostatné informace na vstupu do systému a transformují je na **informace**, které ze systému naopak vystupují. Jednodušeji lze říci, že procesy jsou funkce zabezpečující sběr, přenos, uložení a distribuci informací. Samotné informace jsou tedy **data**, která jsou použita pro rozhodování v širším měřítku systému. Tato data mohou být po zpracování prezentována například koncovým uživatelům tohoto informačního systému. V současnosti je kvalitní informační systém klíčovou částí většiny úspěšných firem v různých oblastech podnikání. Především z důvodu rychlého růstu technické úrovně a neustále

se zlepšujícími technologiím výroby a distribuce produktů a služeb je ve většině případů nutnost mít kvalitní informační systém, který bude pravidelně udržován a postupem času inovován. Právě inovace jedné části informačního systému je popisována v této práci.

V základním měřítku lze informační systémy rozdělit na dva typy: **podnikové** a **veřejné** informační systémy. Primární rozdíl je především v tom, komu daný informační systém poskytuje informace a kdo je může upravovat [5].

### 2.2.1 Podnikové informační systémy

Tento typ informačního systému je využíván především v podniku k podpoře a zlepšování podnikových funkcí a procesů. Tyto systémy provozují společnosti především samy pro sebe, s vlastními daty, ke kterým mají přístup pouze pracovníci firmy [16].

### 2.2.2 Veřejné informační systémy

Tyto systémy uchovávají a nabízejí informace, k nimž má přístup nějaká komunita či přímo veřejnost. Spadají sem například různé systémy pro veřejné knihovny a muzea či případně veřejná správa ministerstva vnitra.

## 2.3 Technologie

V této kapitole budou vysvětleny technologie, použité při návrhu a implementaci dílčí části informačního systému. Jedná se jak o front-end technologie tak i back-end.

### 2.3.1 HyperText Markup Language (HTML)

HTML (*HyperText Markup Language*) je v informatice název značkovacího jazyka využívaného pro tvorbu webových stránek. Webové stránky jsou propojeny pomocí hypertextových<sup>1</sup> odkazů. Tento jazyk je složen z množiny značek (anglicky nazývané **tags**). Tyto značky fungují jako základní stavební kameny všech webových stránek. Značky nejsou navenek viditelné, prohlížeč je v pozadí překresluje do vizuální podoby různých prvků, které jsou prezentovány pro uživatele. Značky se uzavírají do hranatých závorek a jejich pojmenování vychází většinou z anglických slov nebo jejich zkrácených podob. Tyto značky mají také své atributy (anglicky attributes). Tyto atributy následně mohou měnit a upravovat chování vykreslovaných prvků [19].

---

<sup>1</sup>Hypertext - strukturovaný text, jež slouží jako odkaz na jiná místa v systému nebo na jiné systémy

### 2.3.2 Cascading Style Sheets (CSS)

Kaskádové styly (*Cascading Style Sheets*) slouží ke specifikování vzhledu prvků na webových stránkách. Samotné HTML (viz kapitola 2.3.1) dokáže pomocí některých značek v omezené míře nastavit základní vzhled prvků. K definování více rozmanitého grafického nastavení pro webové stránky slouží tedy tyto kaskádové styly. Nastavit tyto styly je možné pomocí různých metod, jednou z nich je jednořádkové definování stylů pomocí atributů v HTML značce<sup>2</sup>. Další možností je identifikovat prvek na stránce pomocí atributu ID<sup>3</sup> nebo atributu třídy (class)<sup>4</sup>. Následně lze vytvořit požadovaný vzhled právě pro tyto třídy nebo identifikátory v separovaném souboru, který může být vložen do stránky nebo ve stránce samotné v hlavičce dokumentu. Stejně jako samotný značkovací jazyk HTML je i CSS zpracováváno na straně klienta. Z toho důvodu záleží na jeho typu a verzi prohlížeče. Některé prohlížeče (především ty starší) vykreslují určité elementy či případně styly jiným způsobem [19].

#### Less

Less (*Leaner Style Sheets*) je rozšíření pro kaskádové styly. Tato knihovna pro kaskádové styly (viz kapitola 2.3.2) zavádí do CSS některé vlastnosti, které jsou známé z jiných programovacích jazyků. Mohou to být například proměnné, výrazy nebo makra. Jak již bylo řečeno v kapitole 2.3.2, pro změnu vzhledu určitého prvku je nutná jeho identifikace a následná změna určité vlastnosti. Pomocí této knihovny jsou všechny tyto změny o poznání snazší. Je možné využít proměnnou do které si vývojář nadefinuje hlavní barvu webové stránky a následně tuto proměnnou použije ve všech CSS souborech a na všech místech, kde tuto barvu potřebuje. Stejně tak je možné použít funkce, které po zavolání vygenerují různé části stylů. Další velkou výhodou této knihovny je možnost zanořování jednotlivých elementů do sebe a díky tomu je samotný kód přehlednější a výrazně zkracuje čas potřebný k vývoji [20].

### 2.3.3 JavaScript

Dalším jazykem, který je, respektive dříve byl zpracováván pouze na straně klienta v prohlížeči, je JavaScript. Technologie tohoto jazyka umožňuje ovládat, detekovat a měnit

---

<sup>2</sup>Definované jednořádkových stylů se podle standardů příliš nedoporučuje

<sup>3</sup>ID - identifikace prvku na stránce, oproti třídě by pojmenování tohoto ID mělo být unikátní

<sup>4</sup>Třída - identifikace prvku na stránce, může být použito vícekrát na jedné stránce

všechny prvky vykreslované do prohlížeče, a to dynamicky po načtení stránky. Je tedy možné detekovat obsah formuláře před jeho odeslání (případně měnit), změnit vzhled různých tlačítek a prvků v závislosti na stavech jiných prvků. Tento skriptovací jazyk je objektově orientovaného typu i přesto, že nezná klasické třídy. Samotné objekty se v jazyku ale nachází a lze s nimi pracovat. Objektem je zda logická jednotka s vlastními stavy zachycenými pomocí atributů a funkcemi zpracovávající tyto atributy. Na začátku této kapitoly bylo uvedeno tvrzení, že JavaScript v současnosti již není zpracováván pouze na straně prohlížeče. Přibližně od roku 2010 začal velký vývoj různých frameworků<sup>5</sup> jako například **Angular**, **React** či **Vue.js**. Tyto frameworky se následně začaly používat i na straně serveru pro generování HTML stránek [37]. Pro snazší použití této technologie, jež nabízí vývoj dynamických a více interaktivních webových stránek, je zde několik knihoven. Jedna z nich bude popsána v následující kapitole 2.3.3.

## jQuery

Knihovna jQuery využívá technologie programovacího jazyka JavaScript, popisovaného v předešlé kapitole. Knihovny obecně ulehčují programování v daném jazyce a zde tomu není jinak. Tato knihovna klade důraz na komunikaci mezi JavaScriptem a HTML. Účelem knihovny je určitá separace selekce prvků, jejich chování a přístup k nim. Mezi první velké ulehčení patří přístup k elementům na stránce a jejich změny (ať už vizuálně nebo obsahově). Snadnější ovládání událostí, snadné využití efektů (skrývání, vysouvání) a různé utility<sup>6</sup>, pomocí kterých lze získat přístup k informacím prohlížeče [15].

### 2.3.4 PHP

PHP (Hypertext Preprocessor) je skriptovací programovací jazyk. Tento jazyk slouží především k programování dynamických internetových stránek a webových aplikací. Provádění skriptů se děje na straně serveru a výsledek je následně přenášen zpět k uživateli do prohlížeče. Tento jazyk není závislý na platformě, je ovšem nutné mít několik systémově závislých funkcí pro podporu interpretace tohoto jazyka (například Apache viz kapitola 2.3.6). Interpret jazyka PHP je také možné volat pomocí příkazového řádku.

---

<sup>5</sup>**Framework** - softwarová struktura, sloužící jako podpora při programování (podpůrné programy, knihovny, podpora pro návrhové vzory apod.)

<sup>6</sup>**Utilita** - pomocný program (nástroj) nebo jeho část, která usnadňuje různé činnosti

Jazyk PHP již ode verze 5.0 podporuje paradigma objektově orientovaného programování<sup>7</sup>. Tento typ programování ulehčuje například přístup k databázi pomocí objektově orientovaných knihoven či přidává možnost více intuitivního a rychlejšího vývoje.

Existuje zde také řada různých knihoven nebo frameworků. Knihovny v mnoha případech pouze přidávají nějakou funkcionalitu. Mohou umožňovat například pracovat s elektronickou poštou (odesílání/přijímání e-mailů), generování dokumentů v různých formátech (PDF, CSV, XLS), či případně export souborů do archívu (ZIP, RAR). Mimo knihoven jsou zde také frameworky. Tyto frameworky ve většině případech mění celkový přístup k programování v daném jazyce a přidávají určitou nadstavbu jazyka [32]. **Nette** je jedním z těchto frameworků a bude popsáno v kapitole 2.3.4

## MVC

MVC (Model View Controller) je spíše softwarová architektura, občas někdy též nazývaná také jako návrhový vzor. Smysl této architektury je oddělit datový model aplikace, uživatelské rozhraní a řídicí logiku do tří nezávislých komponent. Tyto tři komponenty by měly být na sobě nezávislé a jejich přímá modifikace by měla mít pouze malý vliv na celkový běh systému [14].

Jednotlivé prvky MVC architektury

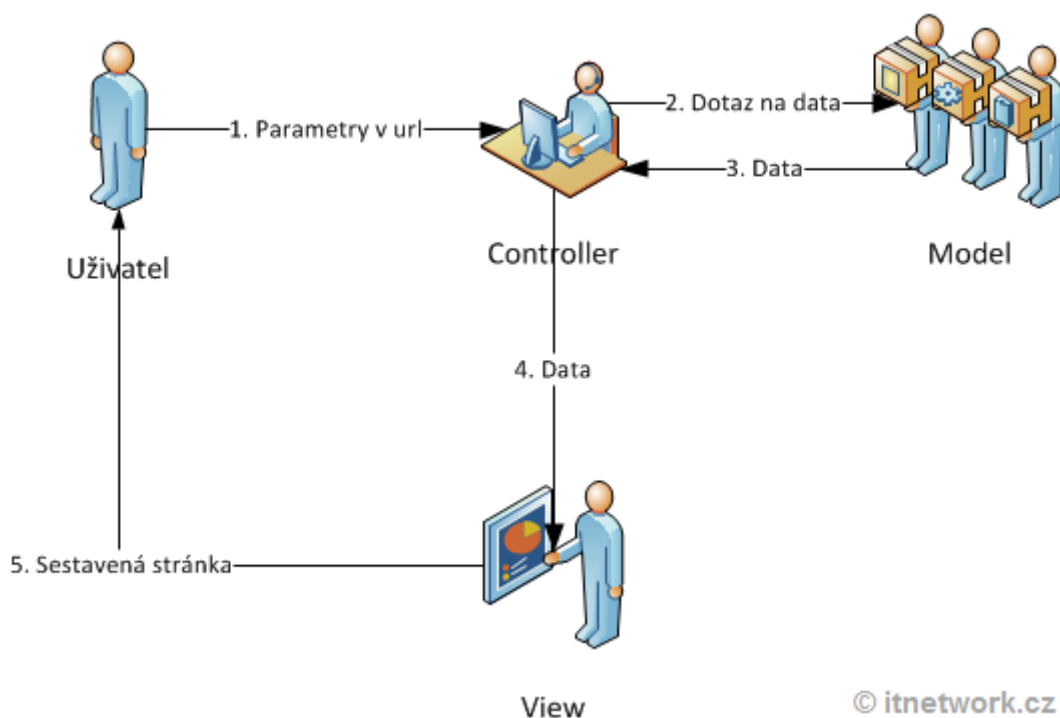
- **Model** (model) - obsahuje logiku a vše, co do ní spadá, výpočty či databázové dotazy [14]
- **View** (pohled) - zobrazení výstupu pro uživatele [14]
- **Controller** (řadič) - prostředník, který řídí komunikaci mezi uživatelem, pohledem a model [14]

Pro lepší představu jak probíhá komunikace v architektuře MVC může posloužit následující obrázek 2.1.

---

<sup>7</sup>**OOP (objektivně orientované programování)** - data a metody jsou zapouzdřeny do objektů, kód je lépe přenositelný a udržitelný, vysoká úroveň abstrakce





Obrázek 2.1: MVC - diagram (zdroj itnetwork.cz) [14]

## Nette

Nette je open source framework<sup>8</sup> pro tvorbu webových aplikací v jazyce PHP pod licencemi GNU GPL a licencí Nette. Využití frameworků obecně má řadu výhod. Ve většině případů částečně eliminují bezpečnostní rizika, která mohou vzniknout v aplikacích a systémech. Podporují různé technologie a integraci systémů, která by byla náročná na konstrukci bez této pomoci.

Tento framework je založený na softwarové architektuře MVC (viz kapitola 2.3.4). Nette obsahuje různé validační skripty, které jsou automaticky generovány. Integrovaný formulářový a dotazovací generátor. Dále má v sobě podporu pro automatické překlady a ochranu proti různým druhům útoků na webové stránky. Nechybí ani možnost tzv routování<sup>9</sup> [29] [32].

<sup>8</sup>**Open source** - volně přeloženo jako otevřený zdrojový kód, znamená, že kód k dispozici zdarma všem, jsou zde samozřejmě určité podmínky pro následné využití za finančním účelem

<sup>9</sup>**Routování** - jedná se o směrování, v kontextu frameworku Nette se jedná o hezké URL adresy které ukazují na jinou stránku než je skutečná adresa

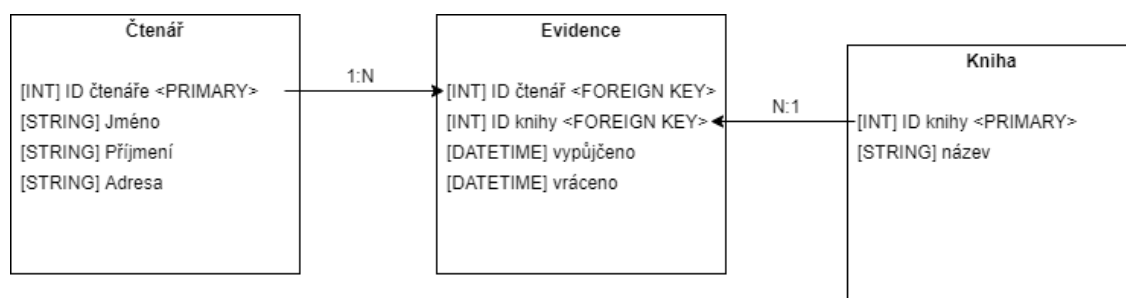
### 2.3.5 Relační databáze

Tato práce se věnuje návrhem a implementace dílčí části informačního systému a zasahuje tedy i do ukládání dat do databáze.

V obecném měřítku slouží databázové systémy k ukládání dat. Relační databáze je založená na relačním modelu. Tato databáze je založená na tabulkách [18]. Jednotlivé řádky tabulky jsou nazývány jako záznamy. Jednotlivé sloupce tabulky jsou atributy. Všechny sloupce (atributy) mají svoje přesně specifikované datové typy (například celočíselné hodnoty, řetězce, čas a datum apod.). Dále, jak již název databázového modelu napovídá, se zde nastavují relace. Tyto relace slouží jako vztahy mezi entitami (tabulkami).

Pro lepší vysvětlení relačních databází poslouží následující příklad viz 2.2. Tento příklad zobrazuje vztah (relaci) mezi jednou entitou **Čtenář** a druhou entitou **Kniha**. Tento vztah se promítá do třetí entity **Evidence**, která se v tomto případě nazývá jako slabá entitní množina. Slabá entitní množina obsahuje několik cizích klíčů z ostatních tabulek. Každý cizí klíč v tabulce Evidence je zároveň primárním klíčem ve své vlastní. Primární klíč by měl být zároveň také unikátním klíčem tabulky. V případě slabé entitní množiny se může jednat také o složený primární klíč, kde lze zkombinovat cizí klíče *čtenáře*, *knihy* a hodnoty *data zapůjčení*.

V tomto příkladu se jedná o systém **knihovny**, kde si některý čtenář může vypůjčit několik různých knih. Ale kde zároveň jedna kniha může být vypůjčena více čtenáři (samozřejmě v jiný čas a datum).



Obrázek 2.2: Příklad návrhu relační databáze (zdroj: vlastní tvorba)

### MySQL a MariaDB

Jedná se o otevřené systémy řízení báze dat, který využívá relační model viz 2.3.5. Komunikace s těmito databázovými systémy probíhá pomocí strukturovaného dotazovacího jazyka SQL. Pomocí tohoto jazyka lze sestavovat dotazy, jež umožňují manipulaci se strukturou

databáze a se samotnými daty. Samotný databázový systém lze snadno nainstalovat na různé platformy (například Linux/Windows). Velkou výhodou těchto databázových systémů je, že je relativně snadné je začlenit do různých druhů aplikací naprogramovaných v různých jazycích [28] [26]. Kupříkladu pro výše zmíněné PHP (kapitola 2.3.4) existuje knihovna **Dibi**, kterou lze integrovat do systému a navázat připojení s MySQL. Mimo samotného připojování je samozřejmě také možné zasílat z jazyka PHP dotazy přímo na databázový server pomocí jazyka SQL. [11].

### 2.3.6 Apache

Apache HTTP server je open source webový server, který je dostupný pro různé platformy (například Linux/Windows). Jedná se o jeden z nejoblíbenějších serverů a nejvíce rozšiřitelných serverů v současnosti. Díky své modularitě, kde lze relativně snadno zapínat podporu a instalovat další součásti, je možné rozšířit statické stránky o dynamické generování obsahu pomocí PHP skriptů. [7]. Pro vývoj a pro většinu programátorů lze využít také aplikace jako LAMP/WAMP, jedná se o kombinace nástrojů Linux/Windows + Apache + MySQL + PHP [4].

### 2.3.7 Git

Git lze v informatice definovat jako distribuovaný systém správy verzí. Tento nástroj je hojně využíván při vývoji software a je jedno ať už se jedná o vývoj ve větším týmu nebo například pokud někdo programuje sám. Při ukládání dat do repozitáře si Git vytváří tzv. snímky rozdílů. Díky těmto snímkům má každý uživatel možnost vidět, co on sám nebo jeho spolupracovníci změnili. Mimo kontroly těchto změn je samozřejmě také možné se vracet ke konkrétním snímkům, tedy stavům projektu uloženého v repozitáři.

Jednou z největších výhod tohoto systému, mimo samotné správy verzí, která umožňuje přechod ke starším verzím, je možnost paralelní práce více členů týmu na jednom projektu. Verzovací systém je jednou z mála možností jak může tým pracovat na jednom projektu či dokonce na jednom souboru v projektu zároveň. Dva lidé provádí různé změny v jednom jediném souboru a následně chtějí tento soubor nahrát zpět do repozitáře. První pracovník nemá žádné problémy a soubory nahraje. Druhému pracovníkovi bude při pokusu o nahrávání souboru řečeno, že repozitář obsahuje novější verzi tohoto souboru. Musí si tedy nejdříve tuto verzi stáhnout k sobě pomocí git příkazů. Následně musí vyřešit případné konflikty. Řešení těchto konfliktů nemusí být nutně nejjednodušší, ale lze snadno

identifikovat, kdo které změny provedl, a tedy jaká by měla být podoba výsledného souboru. Alternativou je využití různých nástrojů, které dokáží automaticky sloučit dvě verze souborů a vyřešit tak tyto konflikty [33].

## **GitLab**

Webová služba běžící pod verzovacím systémem Git. Jedná se o webový nástroj, který umožňuje tvorbu git repozitářů. Každý takový repozitář obvykle obsahuje nějaký projekt nebo jeho dílčí část. Je zde možnost vytvořit také wiki pro daný projekt, sledovat vizuálně změny v projektu a řadu dalších statistik.

### **2.3.8 IPTV**

Jedná se o digitální přenos televizního vysílání za využití internetového připojení. Konkrétně se využívá IP protokol v počítačových sítích. Ve většině případů je IPTV dodávána v uzavřených síťových infrastrukturách. Na rozdíl od klasického televizního přenosu, kde je televizní signál vysílán pomocí světelných impulzů uvnitř optických kabelů, IPTV vykresluje vše skrze internetový kabel. Navíc má koncový uživatel daleko větší možnosti v interakci s tímto vysíláním a může se tedy hodně věcí přizpůsobovat a měnit. Ať už se tedy jedná o samotné pozastavování, přetáčení či případně nahrávání pořadů nabízí tato technologie řadu výhod oproti klasickému vysílání [2].

## **2.4 Projektové řízení**

Jedná se o proces ve kterém jednotlivci nebo případně organizace využívají své zdroje k realizaci projektů. Projekt je činnost, která má jasně daný cíl, začátek a konec a limitované zdroje. Samotná metodologie projektového řízení představuje způsob řízení projektu. Některé metodiky jsou obecně strukturované a lze je přejmout a využít na jiné projekty či případně využít nějakou metodiku šitou na míru potřebám projektu [21].

Poté co je projekt dokončen lze posuzovat a hodnotit různá kritéria. Lze zhodnotit jak kvalitní je výsledek, jehož bylo pomocí projektu dosaženo, čas jež byl vynaložený a náklady. Znalosti kvalitní metodiky projektového řízení jsou předpokladem pro zvládnutí větších projektů, a to především více efektivněji. Mezi další výhody patří také například organizace pracovních týmů.

Projektové řízení lze rozdělit na dva základní přístupy a to **tradiční** a **agilní** přístup. Jak již bylo řečeno, na každý projekt či případně pro každý tým je často třeba využít jiný druh projektového řízení. Jiným způsobem by se řídil projekt výstavby nové budovy a jinak zase vývoj software [17] [27].

#### 2.4.1 Tradiční metodika

Tento přístup k řízení je založený především na přesném plánování hned na začátku projektu a striktním řízením všech aktivit v průběhu projektu. Je vhodné především pro projekty, které mají striktně danou podobu cíle (např. výstavba nové budovy nebo výrobní linky). Je tedy třeba dobře naplánovat a následně řídit všechny aktivity, jejich návaznost či případně některé externí dodavatele. Tento tradiční přístup vyžaduje tedy jasně specifikovaný cíl, výstupy a plán projektu. Tento přístup lze rozdělit do několika fází [25].

- Inicie
- Plánování a návrh
- Realizace
- Monitoring
- Uzávěření

#### 2.4.2 Agilní metodika

Oproti tradičnímu přístupu řízení, je agilní metodika více dynamická. Je založená na průběžném upřesňování cíle díky interakci se zadavatelem projektu (zákazníkem). Tento přístup je vhodný pro takové projekty, kde nelze předem kvalitně specifikovat všechny detaily projektu a cíle. Tento přístup se často využívá při **vývoji software** [25].

Jednou z význačných vlastností této metodiky je především rychlá reakce na požadované změny a možnost se přizpůsobovat novým podmínkám. Existuje několik principů agilního vývoje.

- **Inkrementální dodávky** - jelikož je zákazníkovi dodávána v inkrementálních cyklech pouze část produktu, měla by každá tato část mít nějakou hodnotu, mělo by tedy jít o nějakou ucelenou funkční jednotku, kterou může zákazník vyzkoušet
- **Iterativní postup** - tento postup je definovaný tím, že probíhá pomocí empirického průběhu, postupným zkoušením jednotlivých funkcionalit, jejich předěláváním a zdokonalováním v každé iteraci, díky tomu, že zákazník vidí nahrubo vytvořený projekt, může rychleji dospět k finálnímu zadání

- **Multifunkční týmy** - jelikož se tým soustřeďuje na to, aby projekt dokončil, případně dodal zákazníkovi nějakou část produktu je běžná tvorba multifunkčních týmů, jako například programátoři a testéři pracují spolu, komunikují a udržují se ve vzájemné informovanosti
- **Zapojení zákazníka** - do agilního přístupu je zahrnut také zákazník/klient pro něhož je produkt určen, zákazník přijímá iterativní dodávky produktu ke kterým se vyjadřuje a upřesňuje další požadavky
- **Pravidelná kontrola požadavků (revize)** - jelikož je tento princip řízení nasazován na projekty, kde nemusí být jasně daný rozsah a podoba výsledného projektu, je tedy nutné používat takové postupy, které zabezpečují zpětnou vazbu a průběžnou revizi požadavků
- **Agilní chování** - ne každý tým zvládne agilní vývoj, je nutné aby se zapojil celý tým a pracoval jako jeden, je tedy nutná proaktivita všech členů týmu

## SCRUM

SCRUM je jednou z agilních metodik projektového řízení. Patří mezi nejčastěji používané metodiky. Tato metodika aplikuje základní přístupy agilního manifestu. Dále uplatňuje specifické přístupy řízení vývoje a představuje alternativu k tradičním přístupům tvorby software.

Vývoj pomocí této metodiky probíhá v iteracích (sprintech), které obvykle trvají dva týdny až jeden měsíc, záleží na dohodě týmu, případně vedení. Tyto iterace jsou zakončovány setkáním (lze nazvat také ceremonie). Tohle setkání je většinou složeno ze tří částí: retrospektivní analýza, demo prezentace výsledků a plánování dalšího sprintu.

**Retrospektivní analýza** slouží především k tomu, aby se všichni členové týmu mohli vyjádřit k tomu, co šlo špatně/dobře, co jim nějak vadilo nebo chybělo v uplynulém sprintu. Následuje **demo prezentace**, kde jednotlivé členové mají prostor pro prezentování finálních produktů, případně některých funkčních dílčích částí na kterých pracovali v průběhu sprintu. Ceremonie je zakončena **plánováním**, které se týká plánováním úkolů pro další sprint.

Další velice důležitou praktikou používanou v celém průběhu sprintu je krátká (10 minutová) porada tzv. **stand-up**, celého týmu, která slouží pro informování ostatních členů týmu o průběhu sprintu. Každý člen má zde malý prostor, aby řekl, na čem pracoval mi-

nulý den, na čem aktuálně pracuje, případně bude pracovat a zdali něco/někdo blokuje jeho práci.

Jednou z největších výhod metodiky SCRUM je schopnost efektivně a pružně reagovat na změny, které mohou během projektu vzniknout [17] [27].

## SCRUM - role

V celé agilní metodice SCRUM existují dvě důležité role. Jednou z nich je **scrum master**. Tento člověk je pracovník společnosti, který ovládá metodiku scrum. Je zodpovědný za porozumění a aplikování pravidel této metodiky. Pomáhá členům týmu s komunikací ať už mezi sebou nebo s dalším okolím. Snaží se organizovat různé schůzky a moderuje meetingy. Řada článků uvádí, že by neměl být členem týmu [27].

Další rolí je **product owner**. Tento pracovník kontroluje vývoj práce a definuje vizi produktu. Tento člověk zastupuje cílového zákazníka, pro kterého se daný produkt vyvíjí. Jedná se o článek zastupující filtr mezi týmem a externím okolím (zákazníkem), který požaduje produkt. Spravuje jednotlivé úkoly ve sprintu a výsledně celý produkt.

## SCRUM - ostatní prvky

Další prvky této metodiky jsou v mnoha ohledech spíše volitelné. Není důležité je využívat, ale mohou značně usnadnit celé projektové řízení pomocí této metodiky.

Jedním z těchto prvků mohou být tzv. **story points** (příběhové body). Tyto body přiděluje celý tým agilního vývoje k jednotlivým úkolům, které budou přidány do následujícího sprintu. Přidělování těchto bodů je součástí plánovacího dne jednou na začátku sprintu a jejich hodnota by měla reflektovat objektivní náročnost/obsáhlost daného úkolu. Díky tomu lze určit kolik úkolů a jak náročných do daného sprintu přiřadit [17] [27].

Existuje řada nástrojů, které mohou být nápomocné pro tým a pro samotný sprint. Nástroj **Jira**[3] od společnosti Atlassian. Tento nástroj lze využít pro vizuální tvorbu úkolů, jejich popis, ohodnocování pomocí příběhových bodů, přiřazování pracovníků, kteří budou úkol zpracovávat a spousta dalších možností. Důležitým prvkem je zde také nastavení stavů pro daný úkol. Je nutné definovat pracovní postup pro členy týmu, kterého by se měl každý držet. Například pokud je ve sprintu nějaký úkol, který by si pracovník rád vzal a pracoval na něm, je tedy nutné se k tomuto úkolu přiřadit a změnit jeho stav z *otevřeno* na *rozpracováno*. Následně pracovník využije všechny informace ať už z popisku daného úkolu nebo z komentářů, kde se mohl doptat na různé další věci, k tomu, aby úkol zdárně

dokončil. Po dokončení jeho práce přesune úkol do stavu *připraveno pro testování*. Zde přichází na řadu oddělení **testingu**. V případě, že tohle oddělení neexistuje je nutné, aby product owner zjistil, zdali daný úkol byl zdárně dokončen a následně je přesunut do stavu *hotovo*. Případně je zde možné vyčkat na demo prezentaci, kde pracovník svoji část produktu může prezentovat a může obdržet zpětnou vazbu od product ownera. Díky agilní metodice SCRUM bývá člen z oddělení testingu také v agilním vývojovém týmu pro snadnější a více interaktivní komunikaci.



## Kapitola 3

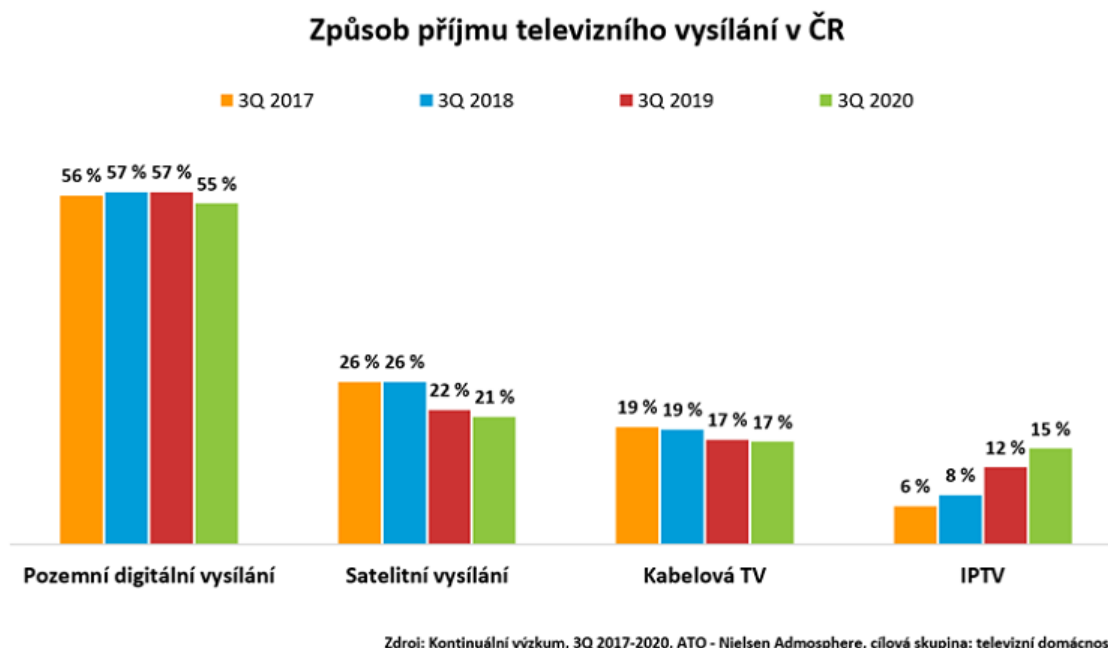
# Analýza problému a současné situace

V této části práce bude popsána společnost na níž bude prováděna řada analýz a následná změna dílčí části informačního systému spolu se změnou projektového řízení. Popis společnosti a její analýza bude proveden do hloubky, jelikož je to nutné pro správné pochopení problematiky. Firma je relativně rozsáhlá, stejně tak jako její systém. Je tudíž nutné porozumět i jemu.

### 3.1 Představení společnosti

Firma, jež je popisována v této práci, se zabývá poskytování internetové televize neboli IPTV. Tato služba umožňuje šíření televizního vysílání za pomoci internetového připojení. Takže jediný důležitý aspekt pro využívání této služby je vlastnit aktivní internetové připojení. Na rychlosti připojení jako takového není kladen nikterak velký důraz, samozřejmě při sledování přenosu ve vysokém rozlišení (FullHD, 4K) by trvalo delší dobu, než se přenos načte.

Sledování televize pomocí internetového vysílání má několik výhod (samozřejmě se počítá se stabilním a kvalitním internetovým připojením). Mezi výhody samozřejmě patří, že není potřeba žádná montáž složitých zařízení, vysílačů apod. IPTV se v poslední době stává více a více rozšířenější (viz obrázek 3.1). Je pravda, že v poměru vůči ostatním televizním alternativám relativně zaostává, jelikož je to docela moderní služba, která si postupem času si získává stále větší oblibu.



Obrázek 3.1: Graf vývoje sledovanosti TV podle technologií (zdroj viz [1])

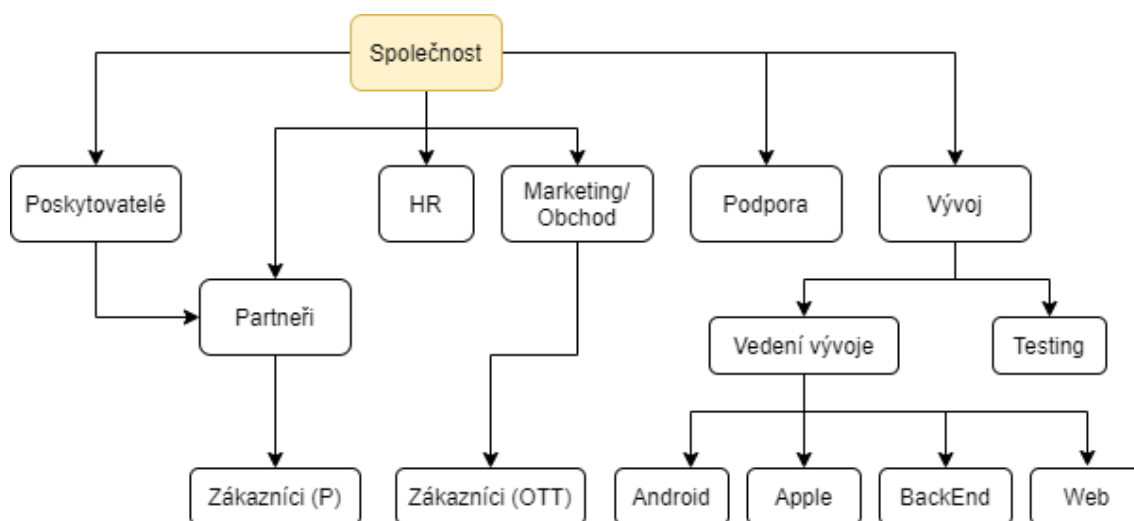
Mimo poskytování samotného televizního a rozhlasového vysílání tato (a vlastně i další podobné služby a společnosti) poskytují také další služby. Mezi ně samozřejmě patří archiv nahrávek, kde je možné si uložit (nahrát) svoje oblíbené pořady, ať už seriály nebo filmy. Případně se lze také dívat zpětně na odvysílané pořady (například 24 - 48 hodin zpětně). V neposlední řadě je tu bohatá a stála nabídka filmů, na které se může uživatel podívat kdykoliv a odkudkoliv.

Tyto všechny výše zmíněné aspekty IPTV a celkového poskytování streamovací služby umisťují společnost do konkurenceschopnosti například s velikány jako je O2 či Netflix. Konkurence na tomto trhu je skutečně velká, proto je třeba se snažit jakýmkoliv způsobem modernizovat a zlepšovat kvalitu svých služeb. Jedním z nejdůležitějších faktorů je samozřejmě spolehlivost a doba tzv. UP-time (doba bez výpadku). Dalším faktorem je výše zmíněná modernizace a snaha o zlepšení vizuální a uživatelské stránky služby.

Popisovaná společnost, mimo poskytování televizního vysílání, vyvíjí také aplikace na různé platformy a druhy zařízení. Dokonce poskytuje na prodej i vlastní SetTopBox pro uživatele, jež nevlastní chytrou TV a chtěli by využít tuhle levnější variantu pro svoji starší televizi. Dále je zde vývoj aplikací na chytré televize, mobilní telefony ať už pro platformy od Apple nebo od Android.

### 3.1.1 Struktura společnost

Společnost popisovaná v této práci je typu B2B i B2C, to znamená, že společnost má svoje partnery (další společnosti), kterým nabízí svoje služby a produkty, ale také má i svoji klientelu koncových zákazníků. Možná přesnějším termínem by se dalo říct, že firma je typu B2C<sup>1</sup> a B2B2C<sup>2</sup>, jelikož služby, které firma poskytuje dalším společnostem jsou pouze upravovány a následně poskytovány dalším koncovým zákazníkům.



Obrázek 3.2: Struktura společnosti (zdroj: vlastní tvorba)

Ve struktuře společnosti na obrázku 3.2 je vidět řada různých vývojových oddělení. Každé z těchto oddělení se soustředí na určitou část systému či případně na konkrétní aplikace. Oddělení Android a Apple se například zabývají vývojem koncových aplikací na mobilní telefony či případně televize. Webový tým se zabývá, jak již název napovídá, především webovou aplikací společnosti. Tato webová aplikace může být spuštěná ve webovém prohlížeči a nabízí všechny plnohodnotné funkce klasické aplikace jako na mobilním zařízení či televizoru. Mimo jiného webový tým spravuje a programuje celou administraci jak hlavní (velká), popsána v kapitole 3.2.2 tak i partnerskou (viz kapitola 3.2.3). Mimo jiné webový tým zajišťuje roli jakéhosi prostředníka (API/Bridge<sup>3</sup>) mezi aplikacemi samotnými a mezi BackEndem. BackEnd se stará o logickou funkcionalitu některých základní částí systémů,

<sup>1</sup>B2C - Společnost poskytující služby zákazníkům

<sup>2</sup>B2B2C - Společnost poskytující služby dalším společnostem, kteří slouží pouze jako prostředník a poskytují služby koncovým klientům

<sup>3</sup>API/Bridge - rozhraní spojující a sjednocující logiku, kterou následně mohou využívat interní i externí systémy přes tzv. žádosti (request)

může se jednat například o pokyn ke generování faktur či zjištění, kde uživatel přestal sledovat určitý kanál nebo film.

### 3.1.2 Partneři a provozovatelé

Jak již bylo dříve zmíněno, společnost je typu B2C ale také i B2B2C. To znamená, že mimo své vlastní koncové zákazníky má i své partnery a provozovatele. Lze si představit, že partneři a provozovatelé jsou vlastně jedna skupina, lišící se pouze v některých detailech. V některých případech se může jednat o provozovatele, který chce svoji vlastní instalaci systému, ať už zahraniční nebo místní. To znamená, že má vlastní možnosti, jak poskytovat IPTV a vlastní hardware na kterém mu bude systém fungovat. Kdežto partneři jsou především lokálního měřítka, veškerá data uchovávají v databázi společnosti a pouze poskytují televizní přenos svým vlastním klientům.

Partneři/provozovatelé jsou ve většině případů zároveň i poskytovatelé internetového připojení. Většinou jsou to lokální poskytovatelé, kteří mají pod sebou menší město, či vesnici. Lidé spadající do jejich sítě mohou využívat služby IPTV zde popisované společnosti pomocí tohoto poskytovatele. Koncoví zákazníci tedy za své služby platí přímo svému poskytovateli zároveň s poplatky za internetové připojení a teoreticky ani nemusí vědět, že za tím vším stojí nějaká větší společnost. Zde popisovaná firma následně fakturuje všechny aktivní přípojky partnerovi. Vzhledem k náročnosti samotného finančního systému a fakturací obecně bude tato problematika popsána v následující kapitole [3.1.3](#).

### 3.1.3 Účetnictví

Samotný systém fakturací a celkově účetnictví je jedním z nerozsáhlejších systému společnosti. Vzhledem k tomu, kolik má společnost partnerů a v kolika různých zemích se tyto partneři nachází, tak musí veškerou fakturaci navrhovat s velkou opatrností vůči všem zákonům dané země a vůči požadavkům partnera. Jak již bylo zmíněno v předchozí kapitole, společnost fakturuje partnerovi veškeré aktivní přípojky a služby, jež využívají jeho koncoví zákazníci. Každý partner má tzv. základní službu, kterou poskytuje svým klientům při registraci. Tato služba má speciální identifikaci na faktuře a je označována jako základní. Všechny ostatní služby se následně účtují jakožto speciální a každá má opět svoji vlastní strategii a různé cenové úrovně, jak se má fakturovat. Hlavní problematika je zde ta, že někteří klienti mohou například svoji službu využívat prvních deset dní v měsíci, následně si službu pozastavit, potom si ji obnovit využívat dalších 5 dní a koncem mě-

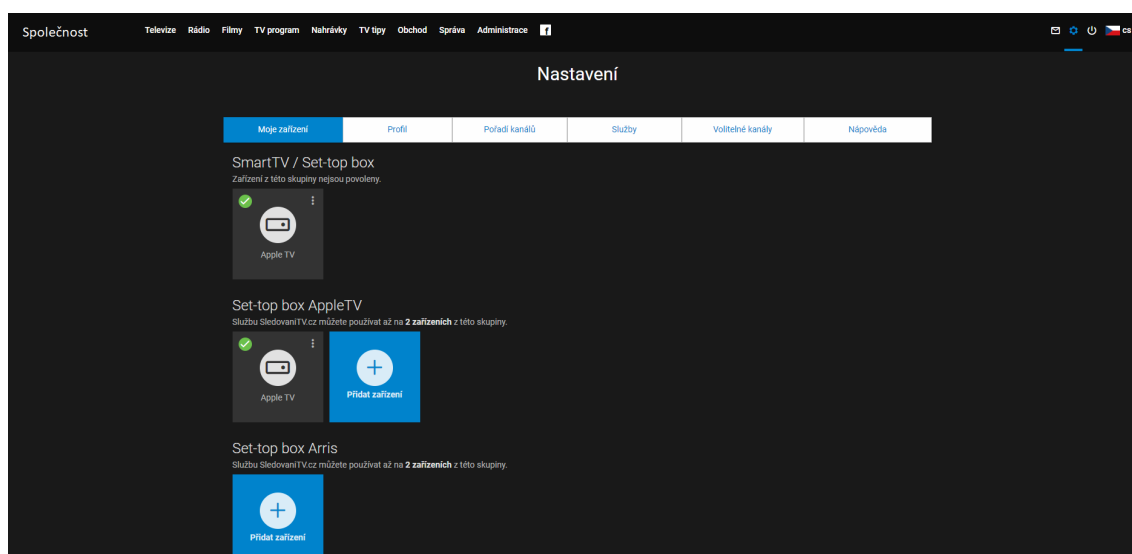
síce opět zrušit a nechat zrušenou. Při původním návrhu systému se s tímto problémem nepočítalo a zákazníkům se fakturovali aktivní přípojky které byly na začátku měsíce. Tato problematika individuálních strategií vyúčtování je velice náročná a nebude rozebrána příliš do hloubky po technické stránce. V jednoduchosti lze říci, že existuje výchozí účetní strategie, která se následně dědí pro každého partnera a podle potřeb se upravuje. Ať už se tedy jedná o speciální poplatky za přípojky, minimální počty přípojek, data vyúčtování a data započítávání aktivních uživatelů.

## 3.2 Informační systém ve společnosti

Samotný informační systém společnosti se skládá z několika důležitých částí. První důležitou částí pro uživatele/klienty je **uživatelská sekce**, veřejná část pro klienty, kde mohou sledovat po přihlášení svoji televizi. Další část je klasická **administrace**, která je poněkud rozsáhlá vzhledem k velikosti systému. V neposlední řadě je tu tzv. **partnerská administrace**, či případně partnerská sekce. A jako poslední část sem lze zařadit **veřejnou sekci**, součástí které může být i internetový obchod, popisovaný v několika kapitolách této práce.

### 3.2.1 Uživatelská sekce

Klienti zde mají možnost sledovat v pohodlí svoji oblíbenou televizi, či případně filmy. V této sekci dále mohou najít různé nastavení, jako zařazení svých kanálů do skupiny *oblíbených*, či různé změny pořadí kanálů, správu platebních karet či případně nastavování rodičovského zámku.

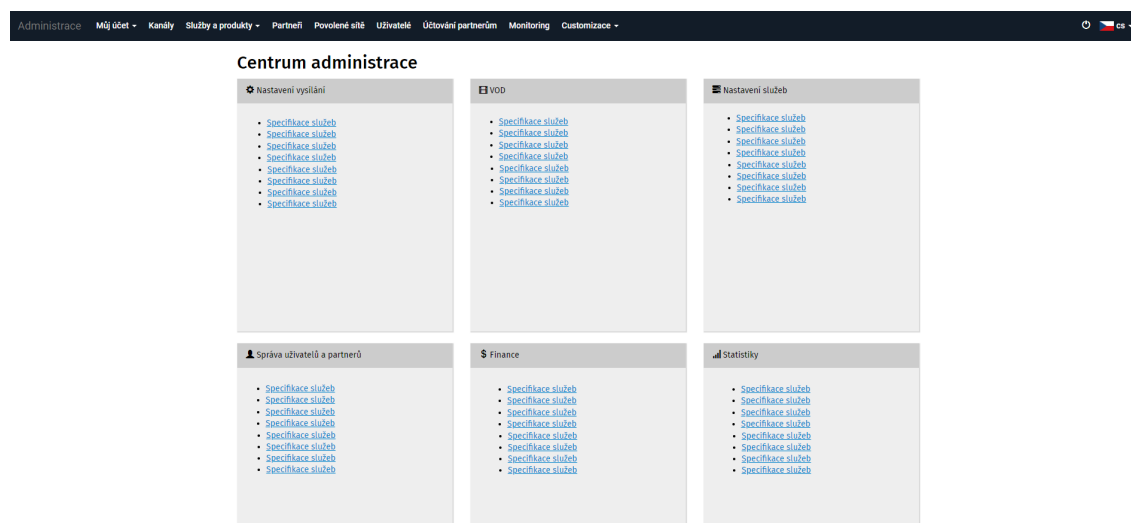


Obrázek 3.3: Uživatelská sekce (zdroj: vlastní tvorba)

Rodičovský zámek bude následně hrát relativně velkou roli ve funkcionalitě nového internetového obchodu. V současnosti je tento zámek 4 číselný kód, který si uživatel může nastavit/změnit v této sekci a tento kód je využíván mimo jiné také pro odemykání kanálů, jež nejsou vhodné pro osoby mladší 18 roků. Je zde nastavená časová platnost pro tento zámek a samozřejmě lze kanály na kliknutí okamžitě zpět uzamknout.

Dále lze zde například párovat nebo odebírat různé zařízení jako například televize, set-top boxy apod. Tato správa umožňuje klientovi mít přehled o všech svých zařízeních, které využívá na sledování televize pod tímto účtem. Nově je zde také přehled všech jeho aktuálně aktivních služeb, které může pozastavovat či případně rušit.

### 3.2.2 Administrace systému



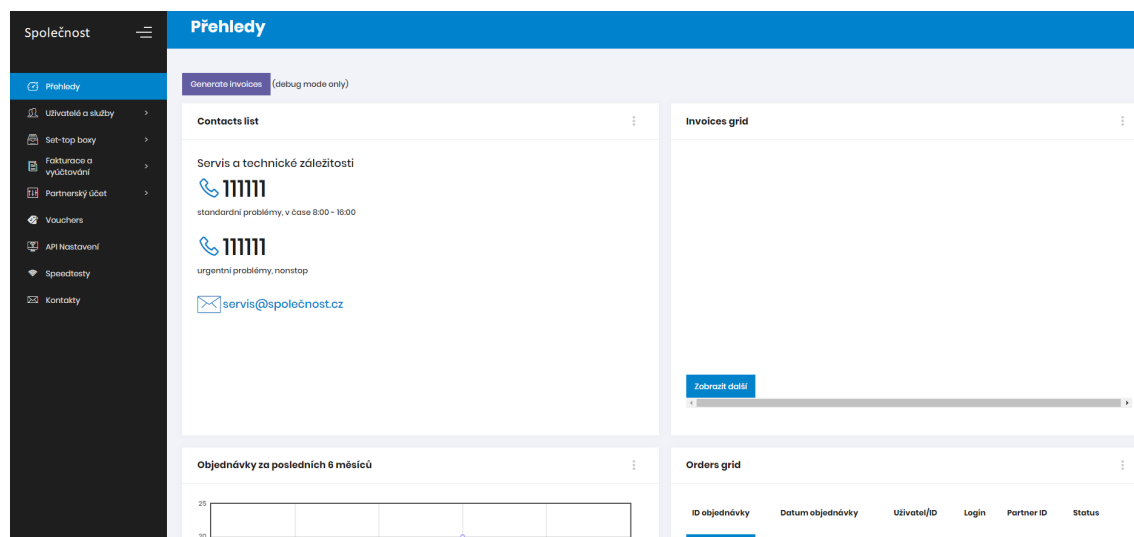
Obrázek 3.4: Administrace systému (zdroj: vlastní tvorba)

Z důvodu částečné cenzury některých věcí, které spadají pod velkou administraci, byly některé položky smazány a nahrazeny jiným textem. Samotná funkcionalita velké administrace zajišťuje veškerou možnou konfiguraci webového systému, jež patří společnosti. Pod touto konfigurací si lze představit všechny věci týkající se obchodního a marketingového oddělení, jako jsou editaci různých ceníků pro všechny partnery, úpravu různých balíčků či konkrétních služeb, kanálů, filmů, rádií apod. Mimo jiného se zde děje samotná editace customizací (více v kapitole 3.2.4). Dále sem spadá veškeré technické nastavení nejenom webového systému, ale také i některých aplikací. Obecně lze říci, že tato administrace zajišťuje konfiguraci naprosto celého systému vyjímaje některých lokálních nastavení v koncových aplikacích. Do této administrace mají přístup téměř všichni pracovníci, kterým jsou přidělovány určité role, které jim umožní přístup do konkrétních sekcí.

### 3.2.3 Partnerská administrace

Tato administrace slouží pro partnery společnosti. Jak již bylo zmíněno v kapitole 3.1.1, tak společnost je jak B2C tak i B2B2C, to znamená, že partneři společnosti mají svoji vlastní administraci, ve které si mohou různě upravovat služby, které jim poskytuje zde popisovaná společnost. Je to spíše takové přeposílání/poskytování jedné služby dále za jiné ceny například. Partner si v této sekci tedy může upravovat svoje vlastní ceníky, do kterých jsou umístěny položky, které by chtěl poskytovat svým vlastním klientům. Tyto

položky jsou vytvářeny ve velké administraci (viz kapitola 3.2.2) a jsou vytvořeny speciálně pro každého partnera. Více informací jak fungují samotní partneři a provozovatelé viz kapitola 3.1.2. V současné době je partnerská administrace rozdělena do dvou variant. Jejich pojmenování je velice jednoduché a to jakožto **stará** a **nová** administrace. Původně se jednalo spíše o grafickou změnu, ale v současnosti je zde i velké množství rozdílných funkcí. Nová administrace je téměř u všech partnerů již aktivovaná a stará bude tedy postupně omezována a rušena.



Obrázek 3.5: Partnerská administrace (zdroj: vlastní tvorba)

### 3.2.4 Přizpůsobení

Název této kapitoly se může zdát poněkud zavádějící, jedná se tedy spíše o tzv. "customizace", což je spíše pracovní název pro takové **profily**. Pod tímto pojmem si lze představit například nějaká barevná schémata webových stránek, které lze velice jednoduše nastavovat. Mimo barev samotných zde lze nastavovat i další možnosti, jako například co se má zobrazovat za jednotlivé položky v menu, co má být viditelné/skryté na různých místech systému a spousty dalších nastavení. Jak již bylo popsáno v kapitole 3.1.1, společnost má svoje partnery, kteří poskytují původní služby svým vlastním koncovým zákazníkům, ale zároveň používají tentýž stejný informační systém jako zde popisovaná společnost. Jediný rozdíl je, že mají vlastní "profil" (customizaci). Tento profil, neboli přizpůsobení, tedy znamená, že každý partner může mít svoje unikátní barevné schéma, logo, skryté například různé odkazy nebo zobrazený internetový obchod i pro veřejnost viz obrázek 3.6. Existuje zde jeden výchozí profil, který používá společnost sama. Z tohoto profilu vychází i všechny



ostatní. Každý partner, který tedy službu tímto způsobem využívá, by měl mít svůj vlastní profil.

Úkony: [Seznam customizací](#) | [Vytvořit nový customizace](#) | [Smazat](#)

## Customizace

[Základní nastavení](#) | [OTT](#) | [Aplikace](#) | [Sekce skrývání](#)

Id\*

default

Název položky\*

Uložit

### Editace nastavení

Hlavní nastavení

Jazyk\*

Čeština

Výchozí jazyk

Unit\*

Výchozí unit

Design\*

Výchozí css styly

Layout\*

@layout.latte

Výchozí layout

Logo\*

Logo v menu

Theme\*

White

Výchozí téma

Odesílatel mailů\*

Mailová adresa pro odesílání mailů ze serveru (např. resetování hesla, ...)

Veřejný uživatel

Veřejný uživatel

Data uživatele (přístupné kanály, objednávání) k zobrazení sekci bez nutnosti

Obrázek 3.6: Ukázka customizace (zdroj: vlastní tvorba)

### 3.2.5 Statické stránky

Tato sekce zahrnuje vytváření statických stránek. Nachází se ve velké administraci a slouží například pro marketingové oddělení, případně pro obchodní oddělení, která vytváří různé reklamy či případně promo stránky. Tyto statické stránky mohou využívat jazyky HTML/CSS (viz kapitoly 2.3.1 a 2.3.2). S využitím těchto jazyků si například člen z marketingového oddělení dokáže vytvořit webovou stránku a následně ji určitým způsobem prezentovat. Mimo samotného programování přes výše zmíněné jazyky je zde možné používat tzv. komponenty. Tyto komponenty jsou interně naprogramovány v systému společnosti a slouží k vykreslování určitých částí, které potřebují mít určitou logiku, která se ve statických stránkách nedá jinak přidat. Může se například jednat o komponentu vykreslení přihlašovacího formuláře či případně některých interaktivních částí stránek, jako volbu jednoho z hlavních balíčků a následné přesměrování do registrace. Tyto statické

stránky lze následně přes administraci prezentovat v několika dynamických částech webu nebo je samozřejmě programátor může vypsát kdekoliv kde chce.

### 3.2.6 Služby

Dříve než bude popsán samotný internetový obchod, je třeba popsat strukturu a chování služeb společnosti. Na těchto službách je založen chod celého systému. Každý uživatel/klient, který chce sledovat televizní vysílání v systému společnosti, musí mít aktivní nějakou službu, která to dovoluje. Bez této služby je okamžitě přesměrován do internetového obchodu či případně na stránku svého partnera nebo poskytovatele. Službu (v systému pojmenovanou jako *Service*) si lze představovat jako nějakou vstupenku, která zákazníkovi umožňuje sledovat některé konkrétní programy, filmy nebo například jenom poslouchat rádia. Samotná služba je velice komplexní prvek systému, a proto je nutné ji rozdělit do čtyř částí, které na sebe postupně navazují.

#### Specifikace

Pod specifikací si lze představit jakousi šablonu, která určuje, co za službu se zde bude prodávat. Uchovává si v sobě typ, který může nabývat hodnot jako například televizní balíček, set-top box, filmy, kvalita apod. Tato specifikace tedy definuje k čemu vlastně bude výsledná služba sloužit. Lze si například určit že specifikace bude zastupovat balíček kanálů/filmů nebo například která umožní klientovi sledovat pořady v ultra vysoké rozlišení (4K).

#### Úroveň

Úroveň služby je přímo závislá na typu služby, a tedy pokud je ve specifikaci zvolen typ služby jako televizní kanály, bude úroveň určovat výběr z balíčků kanálů, které tato výsledná služba klientovi odemkne, například balíček kanálů "History" odemkne několik historických kanálů. Pokud je ve specifikaci zvolen typ "Filmy", bude tato úroveň zastupovat balíčky filmů, jež se klientovi po zakoupení odemknou. Těchto úrovní je relativně velké množství. Další úroveň, kterou je třeba zmínit, je například typ "Další stream", který odemkne volbu úrovně typu 1x, 2x, 5x, 10x. To znamená, že uživatel může sledovat svoji televizi na jednom účtu z více různých zařízení. Pokud tedy si zakoupí balíček se specifikací "Další stream" a s úrovní 5x může pod svým účtem sledovat vysílání až z pěti různých televizí zároveň.

## Balíčky

Mezi balíčky lze zařadit jednak balíčky kanálů, ale také i balíčky filmů. Správa těchto balíčků je velice jednoduchá a jedná se především o "zaklikávání" jednotlivých kanálů pod určité skupiny. Takže v případě balíčku "Sport" budou přiřazeny kanály jako NOVA SPORT, ČT24, ČT SPORT apod. Filmové balíčky fungují na naprosto stejném principu.

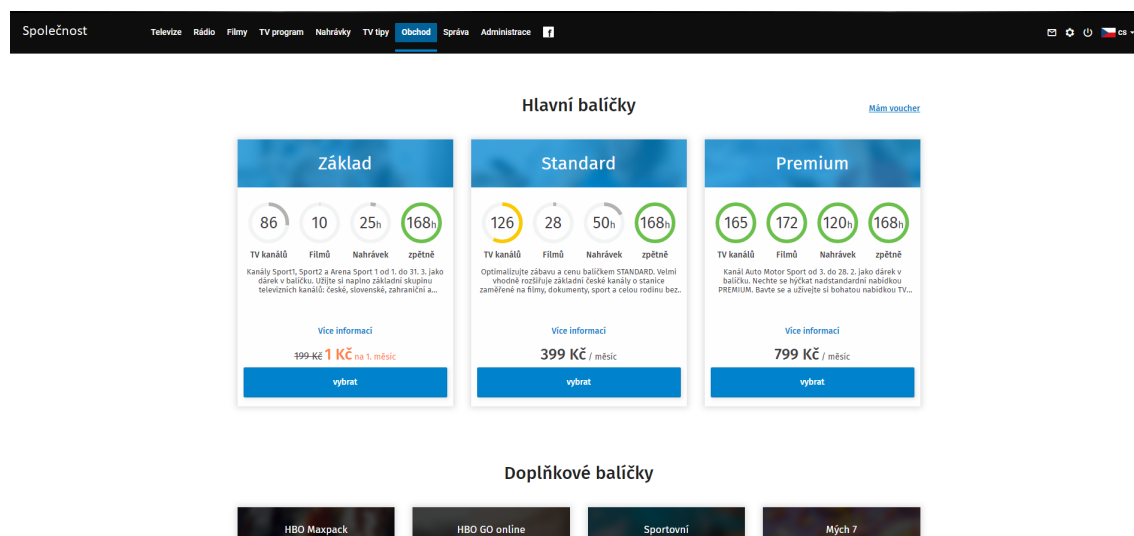
## Položka v ceníku

Položka v ceníku je tedy následně složena ze samotné specifikace, která obsahuje všechno důležité nastavení viz výše. Dále je zde zahrnutá finální cena služby, pořadí a skupinu kde má být tato položka zobrazována a doba po kterou si uživatel může zakoupit tuto službu - zde se jedná především o dobu kdy tato služba bude automaticky obnovována. Také lze zde nastavit různé akční nabídky, jako například výhodnější cena na první měsíc. Ceník dále obsahuje výčet platebních systémů, jež mohou být použity pro platby položek. Stejně tak se zde nachází výchozí platební systém, jež je automaticky zvolen při přidání položky do košíku.

## Finální služba

Ve chvíli, kdy si uživatel zakoupí položku z ceníku, kterou uvidí v internetovém obchodě (viz kapitola 3.2.7), projde úspěšně skrze zvolený platební proces, bude mu aktivována služba. Finální služba je tedy soubor všech předchozích vlastností, které jsou sjednoceny a uloženy jakožto jeden záznam s důležitou informací, a to od kdy a do kdy má konkrétní uživatel zpřístupněny určité funkce.

### 3.2.7 Internetový obchod



Obrázek 3.7: Ukázka současného internetového obchodu (zdroj: vlastní tvorba)

Původní stav internetového obchodu, který slouží k prodeji služeb a produktů společnosti. Vizuální stránka obchodu je vůči dnešním měřítkům relativně dosti zastaralá, což byl jeden z důvodů pro návrh a vývoj nového.

Internetový obchod zde popisované společnosti slouží k poskytování služeb a prodeji produktů. Konkrétněji se jedná o služby poskytování televizního vysílání pomocí internetového připojení (IPTV), jak již bylo dříve zmíněno. Jednotlivé položky tohoto internetového obchodu jsou tedy tzv. balíčky, které kupujícímu zákazníkovi umožní sledovat určitý počet televizních kanálů, určitou kapacitu pro jeho vlastní nahrávky oblíbených pořadů či případně zpětné sledování již odvysílaných pořadů.

## 3.3 Analýzy firmy

V následujících kapitolách budou provedeny některé vybrané analýzy firmy. Tyto analýzy budou sloužit pro nalezení a vyhodnocení problémů, hrozeb, nedostatků a identifikaci sektorů, které je nutné zlepšit/změnit.

### 3.3.1 SLEPTE analýza

Tato analýza je označována jako analýza změn okolí. Umožňuje vyhodnotit případné dopady změn na projekt. Pojednává tedy o vnějších vlivech působících na společnost. Jsou zde

uvedeny faktory z oblastí **sociální, legislativní, ekonomické, politické a technologické** [24].

### **Sociální faktory**

Důležitým faktorem v sociální oblasti je především počet lidí (velikost populace), jež sledují televizní vysílání ať už aktivně či pasivně. Bavíme se především o ČR. Společnost sice poskytuje vysílání i do cizích zemí, pomocí různých partnerů a poskytovatelů, ale ne v tak velké míře jako pro Českou republiku. Nehraje zde příliš roli ani věk nebo pohlaví. Díky relativně nízkým částkám za základní služby je zde i zanedbatelný vliv ekonomické situace působící na klienty, kteří by tyto služby využívali.

Každopádně lze tyto sociální faktory rozdělit do dvou pomyslných skupin, kde jednou jsou tedy koncoví zákazníci, kteří byli popsáni výše a druhou skupinou jsou partneři. Tito partneři pomáhají společnosti s rozšiřováním své působnosti. Partneři poskytují společnosti části zisků za svoje vlastní klienty a částečně se jedná také o reklamu společnosti.

Dostupnost pracovní síly, díky celkovému vývoji světa a díky modernizace technologií, je zde velice dobrá, jelikož lidí, kteří se zajímají o nové technologie a kteří se hlásí na pozici programátorů je relativně velké množství. Například Česká republika a konkrétně město Brno nabízí hned několik univerzit poskytující technologické vzdělání na úrovni. Například jedny z velice dobře hodnocených škol: VUT FIT (Fakulta informačních technologií vysokého učení technického), Masarykova univerzita a Mendelova univerzita. Stejně tak vzdělávací instituty pro tyto lidi vhodné na pozice programátorů jsou na tom velice dobře, je otevíráno stále více škol, které se zaměřují na technické obory. [34]

Díky informacím z Českého Statistického Úřadu (ČSÚ) lze říci, že celková populace České republiky se rok od roku zvětšuje - podle zdrojů až o přibližně 50 tisíc ročně, tudíž v tomto faktoru lze říci, že počet klientů využívající služby společnosti se rok od roku také bude zvětšovat [36].

### **Legislativní faktory**

Pokud jde o legislativní faktory, tak v tomto odvětví nejsou příliš velké změny co by společnost omezovaly, avšak samozřejmě by společnost, byť i ta nejmenší, měla sledovat veškeré nové změny v zákonech. Zákony, jež jsou pro společnost důležité:

- Zákon o obchodních korporacích 90/2012 Sb.
- Zákon o daních z příjmů 586/1992 Sb.

- Zákon č. 586/1992 Sb. Zákon České národní rady o daních z příjmů
- Zákon č. 235/2004 Sb. Zákon o dani z přidané hodnoty
- Zákon č. 231/2001 Sb. Zákon o provozování rozhlasového a televizního vysílání a o změně dalších zákonů

Výše zmíněná společnost má také zahraniční partnery, kterým poskytuje svoje služby, takže musí řešit různé změny zákonů v cizích zemích, jako například Polsko. Různé konverze měn, rozdílné daně apod. [30].

Mimo jiné firma také manipuluje s velice citlivými daty koncových zákazníků, jako jsou například jejich osobní údaje, jméno, příjmení adresa, ale také i informace o jejich kartě a možnost využívat automatické prodlužování balíčků bez dozoru vlastníka karty, to vše spadá do relativně nového zákona o **GDPR** [10].

Následně provádí společnost fakturace svým partnerům, jelikož jak již bylo výše uvedeno firma je také typu B2B2C, je tedy nutné dodržovat pravidla fakturace dle účetních zákonů.

### **Ekonomické faktory**

V současné době lze říci díky statistikám z ČNB, že ekonomika v České republice rapidně poklesává, a to především díky současné pandemické situaci viru COVID-19. S tím vším jde postupně ruku v ruce také dovoz veškeré elektroniky a součástek do výpočetní techniky. Kupříkladu grafické karty ať už středního nebo vysokého výkonu v současné době (únor - březen 2021) na českém trhu vůbec nejsou [35].

Na druhou stranu ale současný stav ekonomiky konkrétně v odvětví služeb poskytující televizní vysílání rozkvétá. Některé články popisují, že rekordy sledovanosti televize jsou snad největší za posledních 17 let. Lze tedy usoudit, že současná pandemická situace a uzamčení většiny obyvatel doma, napomáhá k větší poptávce zrovna po těchto službách [9].

### **Politické faktory**

V České republice aktuální politická situace nikterak příliš nezasahuje do poskytování těchto konkrétních služeb, tudíž tento faktor zde není příliš velkým rizikem. Teoreticky by zde mohla být zmíněna určitá cenzura, která by v budoucnu mohla nastat na některé televizní kanály či případně pořady. Mohla by teoreticky mít za následek částečný pokles poptávky po těchto službách.

Lze se na tento faktor dívat z pohledu partnerů společnosti, kteří jsou živnostníci. Jakožto živnostníci jsou ovlivňováni státním přístupem k malým podnikatelům. V současné pandemické situaci lze říci, že většině malým podnikatelům poskytující služby se nedaří a dařit nebude. Výjimkou zde ovšem tvoří cílová skupina, do které spadá zde popisovaná společnost, která poskytuje služby využitelné v domovech klientů, a tudíž není touto situací negativně ovlivněna.

Některé kanály, jež spadají pod Českou televizi (ČT1, ČT2, ČT24 apod.) podléhají pod státní správu a jsou to vlastně státní kanály. Lze tedy říci, že část televizního vysílání (je to relativně malá část vezmeme-li v úvahu množství všech programů) spadá pod státní kontrolu a lze to tedy zařadit mezi politické faktory [6].

Poskytování televizního vysílání za pomoci internetového připojení (IPTV) je závislé tedy na internetovém připojení každého uživatele. Samotné internetové připojení je v České republice nejrozšířenější pomocí ADSL/VDSL technologie nebo pomocí bezdrátové Wi-fi sítě. Mezi nejvíce využívané poskytovatele internetového připojení lze považovat velké společnost jako O2, Vodafone nebo T-mobile, které využívají, mimo jiné, síť telekomunikací. Tyto sítě spadají pod státní správu, a tudíž jsou svým způsobem na ní závislé [8].

### **Technologické faktory**

V oblasti televizního vysílání a obecně poskytování služeb televizního vysílání pomocí internetového připojení lze zařadit samotné poskytovatele internetového připojení, které je nezbytné pro využívání technologie IPTV. Poskytovatelé internetového připojení lze rozdělit mezi místní poskytovatele, kteří ve většině případů poskytují bezdrátové připojení pomocí různých technologií (např. Wi-fi). Tito poskytovatelé nabízí často internetové připojení pro menší města nebo vesnice. Dále jsou tu poskytovatelé internetového připojení, pokrývající téměř celou Českou republiku (O2, Vodafone, T-mobile). Tito poskytovatelé jsou v současné době nejrozšířenější, jelikož nabízí obrovské pokrytí i v těch nejzazších koutech republiky [13].

Pokud jde o samotné internetové připojení tak jsou zde státní dotace pro rozvoj nové bezdrátové technologie 5G, kterou postupně začíná (respektive chce) využívat například O2.

Lze sem zařadit i neustálý vývoj dalších technologií, které jsou využity k vývoji aplikací, jež slouží k samotnému přijímání televizního přenosu a poskytování dalších služeb. Jako například prodeji různých balíčků, ukládání oblíbených filmů, pořadů apod.

## **Ekologické faktory (environmentální)**

Do těchto faktorů spadá především nakládání s odpady, přístupy k ochraně životního prostředí a celkově věci, které by ovlivňovaly globální ekologický dopad na Zemi. Konkrétně v tomto případě, kde společnost poskytuje především online služby, lze říci, že do toho faktoru příliš nezasahuje. Společnost má své vnitřní prostory v budově JIC v Brně, kde se snaží spolu se svými zaměstnanci o ekologické třídění odpadu do různých druhů popelnic, každá kancelář má svoje tři kontejnery a ve společných místnostech je jich i více.

### **3.3.2 Porterova analýza pěti sil**

Tato analýza patří k základním a zároveň nejlepším nástrojům pro analýzu konkurenčního prostředí společnosti a jejího strategického řízení. Jedná se o analýzu vnějšího prostředí společnosti popisovanou v této práci a poskytující IPTV a její konkurenceschopnosti. Jakým způsobem si má zachovat konkurenceschopnost a různé konkurenční výhody z pohledu pěti faktorů hrozeb [12].

#### **Hrozba stávající konkurence - STŘEDNÍ**

V odvětví IPTV není příliš velká konkurence, jelikož k poskytování těchto služeb je nutné mít smlouvu s tzv. “programmers” což jsou poskytovatelé vysílání samotných kanálů tzv. “streamy”. Jsou zde někteří velikáni, jako například Netflix nebo O2 TV kteří poskytují podobné služby - například společnost O2 poskytuje zároveň i internetové připojení a televizní vysílání lze dokoupit jako další balíček. Dále má společnost O2 také některé exkluzivní programy, které si nikdo jiný dovolit v současnosti nemůže, dle smluvních podmínek.

Další konkurent jako Netflix nabízí poněkud rozdílné služby. Rozdíl spočívá především v tom, že nepodporuje žádné živé vysílání, ale pouze řadu filmů a seriálů z jejich databáze a především je světově hojně rozšířený. Konkurence s touto společností není příliš vysoká z důvodů drobných rozdílů mezi službami. Konkurentem na podobné úrovni může být například **Kuki** nebo **Lepší TV**. Obě tyto společnosti se stále rozvíjí a přichází s novými nápady a lepším/modernějším uživatelským rozhraním.

#### **Hrozba nové konkurence - NÍZKÁ**

V této oblasti je výskyt nový konkurenceschopných protivníků velice malý. Založit a následně vést firmu v tomto rozsahu, která dokáže získat vysílací práva, spravovat několik



různých typů aplikací a udržovat si tuto úroveň vyžaduje nemalé finanční prostředky, a především čas na rozvoj a vybudování důvěry klientů, jež těmto službám budou věřit a budou je využívat.

Na trh samozřejmě může vejít kdykoliv nová společnost, která se bude snažit prorazit, ale v současnosti je trh relativně dost nasycený poptávkou po těchto službách. Je zde tedy vysoká kapitálová náročnost.

### **Hrozba nových substitutů - VYSOKÁ**

Zde v těchto případech je hrozba substitutů relativně vysoká, jelikož mimo samotné IPTV je zde stále klasické pozemní televizní vysílání. Které již před mnoha lety přešlo z analogového na digitální, a tedy i jeho přenos je v dnešní době velice kvalitní. Navíc před nedávnem bylo zavedeno vysílání DVB-T2. Se zvyšující se kvalitou nového pozemního vysílání přichází také překážky jako například nedostatečné technologie (televize/set-top boxy) klientů.

Samotná hrozba je tedy vysoká a IPTV, konkrétně výše zmiňovaná společnost, se snaží kompenzovat tuto hrozbu například poskytováním televizního vysílání na mobilních zařízeních, různě velké databáze filmů a seriálů či případně zpětné sledování pořadů. Některé z těchto funkcí v dnešní době již podporuje i výše zmíněné DVB-T2.

### **Vyjednávací síla zákazníků - VYSOKÁ**

Poskytování televizního vysílání je zde především o klientech a o poskytování vhodných služeb pro ně. Jelikož věk klientů je velice velkého rozsahu, tak se různé společnosti těmto klientům snaží vyhovět pomocí různých kompromisů. Velký důraz je (spíše by měl být) kladen na jednoduché grafické uživatelské rozhraní. Vyjednávací síla zákazníků je zde tedy velice vysoká. Vývojáři do určité míry dbají na zpětnou vazbu od zákazníků a snaží se jim služby často navrhovat téměř na míru. To je také jeden z důvodů proč výše popisovaná společnost nabízí svým klientům velkou řadu různých balíčků. Mohou tedy oslovit například fanoušky sportů, malé děti či případně klienty jež mohou zajímat historické pořady. Také cenové relace mají poměrně velké rozsahy a lze tedy najít konkrétní balíček téměř pro každého.

## Vyjednávací síla dodavatelů - NÍZKÁ

Jako dodavatelé jsou v tomto kontextu myšleni pouze poskytovatelé samotných programů (tzv. *programmers*). Těchto poskytovatelů je velká řada a poskytují různé kanály. Samozřejmě, že získání jakéhokoli nového dodavatele, který zpřístupní některé programy, které aktuálně v nabídce nejsou může být lukrativní, ale naopak absence některého není příliš katastrofická. Jelikož nabídka kanálů je tak rozsáhlá, tak se nestává příliš často, že by klienti kvůli ztrátě jednoho programu přecházeli pod jinou konkurenční službu.

### 3.3.3 Analýza interních faktorů pomocí modelu 7S

Tato analýza hodnotí výše zmíněnou společnost díky sedmi níže uvedeným interním faktorům, které spolu navzájem souvisí [23].

#### Strategie

Strategie společnosti se především týká toho, jak se společnost sama o sobě udržuje v konkurenční výhodě na trhu. Tím dosahuje především poskytováním kvalitních a stabilních služeb/služby, kterou je IPTV (televizní vysílání pomocí internetového připojení). Mimo jiné se snaží vyhovět většině svých klientů k nejvyšší spokojenosti, takže i na samotnou podporu a komunikaci s klienty je zde kladen velký důraz. Jelikož se jedná spíše o poskytování služeb nebo prodej produktů, je tedy vztah mezi poskytovatelem a klientem velice důležitý.

Mimo jiné se společnost rozrůstá do zahraničí a hledá nové klienty v dalších zemích (Polsko, Španělsko). Poskytování těchto služeb v zahraničí vyžaduje samozřejmě smlouvy s jejich vlastními poskytovateli vysílacích programů a další partnery, kteří poskytují internetové připojení v konkrétních zemích. Dlouhodobější cíl je tedy samozřejmě získávání nových klientů, případně nových partnerů, kteří poskytnou další koncové klienty.

Zdokonalování kvality a uživatelské přívětivosti je dalším aspektem na který společnost velice dbá. Jak již bylo zmíněno výše, je služba poskytována veliké škále věkových kategorií, a tedy uživatelská přívětivost by měla být na vysoké úrovni. Mladší generace leckdy ocení moderní a zajímavé rozhraní. Musí být ovšem zachován určitý kompromis kvůli starší generaci, která je často zvyklá na velice jednoduché ovládání, aby po jednom stisknutí tlačítka byla spuštěna televize přesně tam, kde byla předešlý den vypnuta. Strategie společnosti lze shrnout do těchto bodů.

- stabilita 24/7

- uživatelská přívětivost a intuitivní ovládání
- aktivní podpora pro své klienty

## Struktura

V čele společnosti stojí v současnosti především jeden člověk. Oficiálně je zde ještě jeden spolumajitel, který je spíše programátor, než aby řešil nějaké rozsáhlejší organizační věci. Dále jsou tu různé skupiny důležitých oddělení jako například podpora pro klienty, kteří se co nejrychleji snaží řešit všechny problémy a odpovídat na dotazy klientů. Marketingové oddělení a obchodní oddělení, jejichž působení je poněkud rozsáhlé - zabývají se samotným prodejem služeb a produktů, sledování aktuálních trendů na trhu, sledování konkurence, propagování produktů a služeb. Více informací o struktuře společnosti v kapitole **3.1.1**.

Mimo samotného poskytování služby IPTV se společnost zabývá také vývojem různých druhů aplikací na různé platformy (Android, Apple, web, STB). Každou aplikaci vyvíjí samostatné oddělení složené z několika programátorů. Samozřejmě je zde také důležitý návrh uživatelského rozhraní z pohledu UIX a následný samotný grafický návrh.

Dále společnost nabízí také prodej samotných set-top boxů (vyrobeny přímo společností) a to především pro starší generace klientů, kteří ocení co nejjednodušší ovládání a co nejplynulejší zapínání svých oblíbených pořadů.

## Systémy řízení

Mezi systémy řízení zde mohou například patřit organizační pomocné prostředky jako **Google kalendář**, kde je možné sledovat vytíženost a aktuální stav pracovníků, aby například bylo poznat, zda mají dovolenou, jsou v kanceláři nebo na home office. Dále je využit nástroj **Google Meet**, který spolupracuje s Google kalendářem a kde se dají pořádat různé schůzky a plánování dalších aktivit. Dalším systémem může být rozsáhlý softwarový nástroj Jira od firmy Atlassian, jež zde slouží pro všechny vývojové týmy pro organizaci jejich práce.

Právě systémy řízení jsou ve společnosti nejméně rozsáhlé a je zde velký prostor pro zlepšování. Vývojový oddělení v současné době bohužel nedisponuje žádnou efektivní metodou projektového řízení a tudíž samotný vývoj je často velice zdlouhavý.

## **Styl řízení**

Ve společnosti vládne demokratický styl řízení. Zaměstnanci se mají možnost vyjadřovat k různým otázkám organizaci či vedení. Pravděpodobně za to může také to, že firma byla založena programátory, a tudíž se dbá i na jejich názor. Samozřejmě majitel firmy a vedení různých vývojových týmu má vyšší rozhodovací právo, ale současně i dbá na názory ostatních.

Bohužel rozhodování v tomto demokratickém stylu řízení je leckdy velice zdlouhavé, a ne vždy přináší kladné výsledky.

## **Sdílené hodnoty**

Vzhledem k tomu, že společnost je relativně malá, pokud jde o počet zaměstnanců, tak je atmosféra spíše přátelštější a celkový vztah nadřízený vs podřízený je spíše neformálnějšího stylu. Především díky této atmosféře lze říci, že zaměstnanci dokážou táhnout za jeden provaz a efektivně řešit společně problémy a samozřejmě také problémy svých klientů. To znamená, že i například sám programátor může kdykoliv odpovědět na potřeby koncových zákazníků a může například pomoci vyřešit problém, který není oficiálně v jeho popisu práce.

## **Schopnosti**

Jak již bylo zmíněno v odstavcích výše, tak firma byla založena programátory, jež měli velkou zkušenost s vývojem aplikací a webových nástrojů. Mimo jiné jeden z nich měl i velké zkušenosti s marketingem a měl kvalitní organizační schopnosti, takže se jim podařilo velice rychle rozrůstat. Jednotliví členové se díky přátelské atmosféře a velice dobrými vztahy na pracovišti zabývají nejen problémy, jež mají v popisu práce, ale především se snaží pomoci svým kolegům. Mimo jiné jsou dobře motivováni ke zdokonalování svých znalostí. Společnost pro své zaměstnance poskytuje také řadu přednášek a možností pro sebevzdělávání.

Mimo jiné může kterýkoliv programátor během spolupráce se společností změnit i konkrétně svůj tým, a tedy i samotný programovací jazyk ve kterém ve firmě začal a může takto rozšířit své znalosti, setkat se s vývojem jiných/nových technologií a samozřejmě tím i zvýšit svoji hodnotu pro budoucí možnosti na trhu práce.

## Spolupracovníci

Tato kategorie již byla částečně popsána v kapitole *Schopnosti a Sdílené hodnoty*. Pracovníci jsou samozřejmě motivováni velice dobře platovým ohodnocením, které je v tomto oboru relativně vysoké. Mimo jiné mají také příslib a spousty možností ohledně zdokonalování svých znalostí a schopností. Jsou jim nabízeny různé přednášky, konference týkající se nových technologií apod. Mimo jiné mají všichni zaměstnanci k dispozici vybavenou kuchyňku, do které je každý týden poskytovány různé druhy jídel od ovoce, zeleniny po různé sladkosti, cereálie apod. Dalším benefitem mohou být mimopracovní aktivity jako teambuilding nebo Xbox či virtuální realita.

### 3.3.4 SWOT analýza

Díky této analýze budou vyhodnoceny slabé a silné stránky společnosti a jejich služeb a produktů. Jedná se o takový souhrn situačních analýz vnějšího a vnitřního prostředí společnosti [31].

Vnitřní prostředí	
Silné stránky	Slabé stránky
<ul style="list-style-type: none"><li>• Vedení zkušených programátorů</li><li>• Přátelské vztahy zaměstnanců</li><li>• Znalosti zaměstnanců</li><li>• Dobrá motivovanost zaměstnanců</li><li>• Dobré jméno společnosti na trhu</li><li>• Technologické vybavení společnosti</li></ul>	<ul style="list-style-type: none"><li>• Některé techničtější problémy</li><li>• Slabá úroveň projektového řízení</li><li>• Organizační schopnosti</li><li>• Příliš velký rozsah systému (projektu)</li></ul>
Vnější prostředí	
Příležitosti	Hrozby
<ul style="list-style-type: none"><li>• Pandemická situace ve světě</li><li>• Nové stanice pro vysílání</li><li>• Rozšiřování pokrytí internetového připojení</li><li>• Velký nárůst sledovanosti TV</li><li>• Cílení reklam na konkrétní skupiny</li><li>• Velký zájem nových zaměstnanců</li></ul>	<ul style="list-style-type: none"><li>• Stávající konkurence</li><li>• Technické problémy - externí</li><li>• Zrušení smluv od poskytovatelů programů</li><li>• Využití externího frameworku</li></ul>

Tabulka 3.1: Swot analýza

#### Silné stránky

Mezi silné stránky lze především započítat znalosti programátorů ve společnosti. Firma nabízí obrovský potenciál ve vzdělávání a celkově v získávání zkušeností pro programátory. Díky tomu, že firma je stále relativně malá, tak nabízí velice neformální a přátelský přístup a dává tedy možnosti i nepřiliš zkušeným programátorům. Ti se většinou velice rychle začlení do týmu a události jim nedají jinou možnost než se rychlým tempem vzdělávat, aby stačili svým kolegům.

Zaměstnanci jsou také velice dobře motivováni, mimo samotného velice dobrého platového ohodnocení (konkrétně u programátorů) mají i další spoustu benefitů. Tím mohou být například různé členské karty pro sportovní aktivity, stravenky, sick days, velice dobré vybavení kanceláří a možnosti dokoupení různých nástrojů, které by mohly zjednodušit výkon práce pro zaměstnance. Společenské prostory, kde je k dispozici Xbox a virtuální realita, vybavená kuchyňka, kde je dostupné jídlo pro zaměstnance.

### **Slabé stránky**

Mezi slabé stránky lze počítat i technické problémy, což se na první pohled může zdát, že koliduje s jedním bodem v silných stránkách, kde společnost poskytuje různé zařízení zaměstnancům. Tyto technické problémy jsou především neočekávané a může se například jednat o různé výpadky služby, které nejsou nutně zapříčiněny společností samotnou, ale lze jim určitým způsobem předcházet a pojistit se proti těmto událostem. Může se například jednat o nedostatečnou rezervu kapacity na úložištích společnosti, která se zdá být dostatečná, ale při problémech se může neočekávaně zaplnit, a tudíž než programátoři stačí detekovat problém, služba může kvůli této kapacitě přestat fungovat.

Mezi další slabé stránky patří organizační schopnosti a celkové problémy projektového řízení, které budou také hlavním bodem dalších kapitol. Společnost není příliš velká, čítá okolo 70 zaměstnanců, což je podle měřítek spíše menší podnik. Ale i přesto má firma velké množství oddělení, které spolu musí spolupracovat na řešení různých problémů. Proto je nutné, aby firma měla fungující organizační strukturu a řadu pravidel sloužící pro identifikaci problémů, vyhledání vhodného postupu k řešení problémů, vyhledání vhodného týmu, který bude problém řešit a následně samotné projektové řízení v konkrétním týmu, který bude problém řešit. Aktuální stav projektového řízení je bohužel takový, kde si každý, kdo je nějakým způsobem nadřízený některého pracovníka, může dojít a zadat mu nějaký konkrétní úkol. V této chvíli je ale pracovník samozřejmě na pochybách, zda pokračovat ve svém stávajícím úkolu nebo zdali prioritizovat úkol, který mu byl aktuálně zadán. Podobných problémů je nespočet, především díky tomu, že zadavatel úkol nemá přehled o tom, co zaměstnanec konkrétně řeší, nemá ponětí o tom jak bude jeho úkol časově a technicky náročný a tudíž tím vnese chaos do celkové práce konkrétního programátora.

## Příležitosti

Mezi příležitostmi lze zařadit i aktuální pandemickou situaci ve světě. Na první pohled tento faktor nezní příliš pozitivně, ale díky aktuální situaci ve světě a konkrétně i v České republice, kde většina občanů tráví čas doma a krátí si chvíli sledováním televize, je tohle obrovská příležitost pro společnost, která toto televizní vysílání poskytuje. S tím jde ruku v ruce i celkový nárůst uživatelů využívající služby IPTV. Díky dobrému jménu firmy a celkovému dobrému první dojmu firmy, má o práci ve společnosti zájem velká řada pracovníků.

## Hrozby

Mezi hrozby lze zařadit například stávající konkurenci, která byla zmíněna i v *Porterově analýze* v kapitole 3.3.2. Hrozba stávající konkurence byla označena jako střední. Na trhu, kde společnosti nabízí IPTV je několik velikánů, jak již bylo zmíněno dříve, jedním z nich je například **O2 TV**, tato společnost se teoreticky řadí mezi konkurenty, avšak neposkytuje úplně stejné služby jako zde popisovaná společnost. O2 TV je obecně i poskytovatel internetového připojení pomocí různých technologií ať už bezdrátově či případně pomocí ADSL/VDSL. Díky tomuto faktu ve většině případů, kdy si lidé nechají zřízovat internetového připojení, tak si zároveň dokoupí televizní balíček. Samozřejmě jim nic nebrání zřídit si pouze internetové připojení a následně si televizní balíček koupit u jiné společnosti.

Lze tedy říci, že svým způsobem je konkurence, a tedy i hrozba společnosti, například Netflix. Tento poskytovatel filmů a seriálů z Ameriky ale neposkytuje žádné živé vysílání. Nelze tam tedy sledovat žádné online přenosy sportovních utkání či jiné věci. Díky tomu, že Netflix nabízí řadu filmů a seriálů a jak již bylo zmíněno dříve, stejnou funkci nabízí i zde popisovaná společnost díky svému archivu a zpětnému sledování, tak je tento velikán také zařazen do hrozeb a konkurence společnosti.

Na trhu jsou i další společnosti, které lze už považovat za ekvivalentní konkurenty. Například internetové televize **Kuki**. Tato společnost nabízí víceméně stejné služby jako zde popisovaná společnost, ale spíše klade důraz na vizuální stránku služeb než na technickou, není tedy tolik spolehlivá jako konkurenční služby.



### 3.3.5 Výstupy analýz

Z předchozích analýz lze určit, že firma by se měla zaměřit na zlepšení dvou věcí. Jednou z nich je samozřejmě inovace a zlepšování vlastního systému pro udržení dobré pozice na trhu a konkurenceschopnosti a druhou z nich je pokulhávající projektové řízení, které jde samozřejmě ruku v ruce se samotným vývojem. V současné době je úroveň projektového řízení společnosti na velice nízké úrovni.

První věcí je inovace webového obchodu sloužící k prodeji balíčků, a tedy k samotnému poskytování služeb, jež je pro společnost nejdůležitější částí systému. Vzhledem k analýzám, lze říci, že vyjednávací síla zákazníků je vysoká, a tudíž se klade velký důraz na to, aby interakce uživatele se systémem byla co nejhladší a nejintuitivnější. Zajistit klientovi rychlý, bezpečný nákup či případně výběr vhodného balíčku přímo na přání klienta by byl velký krok kupředu a rozhodně by společnosti prospěl. Obecně inovace uživatelské sekce, se kterou je zákazník neustále ve styku je nezbytnou budoucí úpravou systému.

Druhou částí problematiky, které z předchozích analýz vychází je nedostatečná úroveň projektového řízení ve společnosti. Tento problém se týká především vývojového oddělení. Správně navržené projektové řízení je nezbytným aspektem prosperující a úspěšné společnosti. Aktuální stav ve společnosti poukazuje na to, že zde vládne demokratický styl řízení. Tento styl řízení by obecně nemusel být na škodu, ale bohužel nedostatečná poslušnost vedení, špatně odhadnuté časové milníky pro splnění úkolů, nekvalitně vypracovaná zadání úkolů či případně samotná komunikace mezi členy týmu nebo mezi týmy samotnými, vede k tomu, že práce je dodávána se zpoždění nebo případně není dodána vůbec.

## Kapitola 4

# Návrh řešení a jeho přínos

### 4.1 Aktuální stav

Současný stav stávajícího systému, respektive dílčí části systému - webového obchodu je popsán v kapitole 3.2.7. Stav aktuální projektového řízení byl vyhodnocen jako špatný a to po provedení a vyhodnocení několika analýz, konkrétně v kapitole 3.3.5.

Vzhledem k aktuálnímu stavu internetového obchodu společnosti bylo usouzeno, že je nejvyšší čas k jeho inovaci. Aktuální vzhled tohoto obchodu je viditelně zastaralý a nepřilíš vizuálně přívětivý. Orientace v něm pro uživatele není sice příliš náročná, ale ani je nikterak nemotivuje ke koupi různých služeb a produktů. Po provedených průzkumech, dotazování některých klientů, bylo vyhodnoceno, že zákazníci obchod sice využívají ke koupi nebo vylepšení některých jejich stávajících služeb, ale že z toho nemají příliš dobrý pocit. Ani samotný proces platby není příliš přívětivý. Dalším velkým problémem je jednotvárnost položek v obchodě. Každý balíček, jež je nabízen k prodeji, vypadá naprosto identicky jako všechny ostatní. Chybí zde tedy možnost přizpůsobení jednotlivých prodejních balíčků, a to jak samotnou barvu balíčku, obrázek na pozadí či různé další úpravy.

Spolu se špatným stavem tohoto webového obchodu byla vyhodnocena také nedostatečná úroveň projektového řízení firmy, a to především organizace, zadávání nové práce a komunikace mezi členy týmu nebo mezi týmy samotnými. Současný stav projektového řízení, jak již bylo zmíněno dříve, je takový že zde žádné projektové řízení není. Zadání práce je velice sporadické a neorganizované, stejně jako samotný vývoj některých částí systému. Nedostatečná komunikace, špatně formulované zadání s nedostatkem podkladů nutných pro vývoj vede tedy k nedodržení lhůty pro vyhotovení určitého projektu nebo při rapidních rozdílech mezi očekáváním vedení a reálným výsledkem.

## 4.2 Požadavky

Hlavním požadavkem zde bude hrát roli inovace, respektive návrh vývoj nového webového obchodu pro společnost za využití nového projektového řízení. Náhled a popis současného stavu obchodu lze vidět v kapitole 3.2.7. V současné době je tato stránka vizuálně někde na středních hodnotách. Nevyužívá příliš moderního vzhledu, obsahuje relativně velké množství informací a jednotlivé položky jsou víceméně identické, takže si uživatel nemůže vizuálně zvolit kterou položku si koupit. Vzhledem k tomu, že většina klientů v obecném měřítku nakupuje především očima a až poté čte popisky a různé další detaily o produktu, je proto nutné uskutečnit rozsáhlé vizuální změny.

- kompletní změna vzhledu webového obchodu - modernější černý design
- možnost pro obchodní oddělení přizpůsobovat všechny balíčky v administraci
- reklamní plochy s ohledem na to, co klient potřebuje nebo rád využívá

Dále je nutné myslet také na to, že určití partneři a provozovatelé chtějí například využívat stále současnou verzi webového obchodu a nechtějí přejít na novou, ať už z důvodů obav, že nová verze nebude dostatečně vyladěná a spolehlivá nebo se případně nová verze nehodí do jejich celkového vzhledu webových stránek. Z tohoto důvodu je nutné zahrnutou většinu přizpůsobení obchodu do **customizace** (viz kapitola 3.2.4). V tomto ohledu by mělo jít nastavovat následující možnosti.

- aktivovat/deaktivovat novou verzi webového obchodu
- měnit pořadí některých statických sekcí, například hlavní balíčky
- možnost aktivace/deaktivace statických reklam
- možnost přiřazovat statické stránky (kapitola 3.2.5) do bloků pro statické reklamy (bannery<sup>1</sup>)
- aktivace/deaktivace dynamických reklam podle historie uživatele

Další částí, na kterou je potřeba myslet je také samotné projektové řízení, bez kterého by provádění jakékoliv změny zabralo několikanásobek času nebo by případně ani nebylo dokončeno.

---

<sup>1</sup>Banner - česky lze přeložit a jednoduše vysvětlit jako reklamní proužek určený pro webové stránky

Vzhledem k tomu, že současné projektové řízení je na velice nízké úrovni, je zde spousta možností s čím pracovat a jak navrhnout projektové řízení úplně od nuly a efektivněji. Nové projektové řízení bude postupně integrováno již během návrhu a vývoje dílčí části informačního systému popisovaného v této práci. Nové projektové řízení se bude týkat především vývojového oddělení a konkrétně webového týmu. Bude se jednat o zavedení metodiky **Agile** (viz kapitola 2.4.2) a konkrétně **SCRUM** 2.4.2.

## 4.3 Lewinův model řízení změn

Třífázový model změny patří mezi nejvýznamnější modely změn v organizaci. Změna je rozdělena mezi tři fáze: rozmrazení, vlastní změna, zmrazení [22].

### 4.3.1 Síly působící na změnu

Cíle, které by měly být dosaženy pro úspěšné zavedení změny do společnosti, respektive pro úspěšný návrh a implementaci webového obchodu za využití nového projektového řízení.

- moderní a intuitivní webový obchod
- zvýšená úroveň projektového řízení
- jasná struktura řízení týmů
- zvýšení tržeb a rychlost vývoje dalších dílčích částí

Na změnu, která by měla být aplikována působí různé síly. Tyto síly jsou ohodnoceny v rozmezí od **-10 bodů** do **+10 bodů**. Kladné hodnoty jsou pro síly, které podporují tuto změnu a negativní hodnoty identifikují síly, které jsou proti změně.

Následující tabulka ukazuje podstatné síly, které působí buď PRO nebo PROTI změně a jejich ohodnocení.

Síly působící <b>PRO</b> změnu (+28)		Síly působící <b>PROTI</b> změně (-15)	
programátoři - rozmrzelí a nespokojení	+9	finanční náklady	-5
majitel	+6	časová nákladnost změn	-6
zvyšující se poptávka po službách	+8	rozsáhlejší restrukturalizace řízení	-4
zlepšení efektivity týmu	+5		

Tabulka 4.1: Síly působící na změnu

Z tabulky výše je tedy jasné, že **kladné hodnoty** pro realizaci převažují, a tedy navrhované změny mohou být aplikovány. Pokud jde o programátory, kteří jsou PRO změnu, může to být poněkud zavádějící. Obecně jsou **programátoři** velice ochotní a vstřícní, pokud jde o jakékoliv zlepšení současné práce a zlepšení projektového řízení. Samozřejmě se najdou ovšem tací, kteří neradi podstupují některé změny a chtěli by mít pouze okamžité výsledky bez nutnosti se jakkoliv vzdělávat nebo se snažit zapojit do těchto změn. Majitel je také v poněkud rozporuplné pozici především kvůli zvýšení finančních a časových nákladů, pozdržení některých zakázek z důvodu zavedení nového projektového řízení. Ovšem vývoj nového internetového obchodu majitel společnosti podporuje a vlastně i tuto změnu inicioval.

#### 4.3.2 Agent změny

Odpovědná osoba, která má na starost celý proces změn/y. V případě zde popisované společnosti a popisované změny tím člověkem bude vedoucí vývojového oddělení. Tento pracovník má na starost všechny vývojové týmy a teoreticky by se měl starat také o komunikaci mezi nimi. Této osobě bylo od partnerů nebo provozovatelů poskytnuto určité zadání a potřeby, které by rádi ve svém systému měli. Mimo jiné je tato osoba také v pozici, kde bude moci zavést novou metodiku projektového řízení.

Díky nadcházející změně se vedoucí vývojového oddělení bude snažit o poněkud jinou strukturu řízení. Agent změny je tedy seznámen s veškerým děním okolo jednotlivých týmů, které budou do změny zahrnuty a s jednotlivými vedoucími týmů. Zároveň agent změny ví moc dobře co se bude za změnu provádět a v jakém rozsahu. Během procesu změny bude konzultovat aktuální stav s externí společností, která již tento styl projektového řízení má zavedený. Agent změny bude také informovat majitele společnosti o aktuálním stavu procesu změny.

### 4.3.3 Intervenční oblasti

Jednou ze zasažených oblastí je sám majitel společnosti, který bude financovat všechny změny, a to jednak samotný webový obchod, ale také i novou metodiku projektového řízení. Mimo majitele jsou zde další oblasti, které budou zasaženy. Organizační struktura firmy je zasažena nejvíce, spolu s ní bude ovlivněna také technologie firmy a komunikační toky. Obě tyto oblastní budou popsány v následujících kapitolách.

#### Organizační struktura firmy

Změna se zde promítne v docela velkém měřítku, jelikož budou vytvořeny další dvě nové pozice, jednou je tzv. “Product owner” a druhou “Scrum master” viz kapitola 2.4.2. Na jednu z těchto dvou pozicí bude pravděpodobně přidělen vedoucí každého týmu. Celková struktura bude tedy změněna, a to i v ohledu zadávání práce programátorům a formulování samotných zadání.

#### Technologie firmy

Oblast technologie firmy bude též ovlivněna, a to hned ze dvou úhlů. Prvním je implementace samotného webového obchodu, respektive jeho kompletní inovace a změna. V další části se jedná o technologie využívané v projektovém řízení. Nástroj jako například **Jira** bude využit na organizaci práce, kde budou vizuálně vytvářeny tzv. **sprinty**. Tyto sprinty jsou časové rozsahy o dvou týdnech ve kterých bude jeden tým pracovat na řadě definovaných úkolů a na ničem jiném. Mimo jiné budou také ve větší míře využity nástroje **google kalendář** a **google schůzky**, kde se budou každý den pořádat tzv. *stand-upy*. Více informací v kapitole 2.4.2.

#### Komunikační a organizační toky a procesy firmy

Tato oblast bude také ovlivněna, a to především v ohledu komunikace zadavatele úkolu vůči programátorům především kvůli změně projektového řízení. Zde bude komunikovat především programátor (vývojář) se svým Scrum masterem nebo Product ownerem pomocí nástroje **RocketChat** a pomocí komentářů v aplikaci **Jira**. Jednou za dva týdny se sejde tým na společnou “ceremonii” (viz kapitola 2.4.2, kde je součástí i plánování následujícího sprintu. Většina důležitých informací o úkolech bude sdělována písemně, aby všichni členové týmu měli stejné informace.

#### 4.3.4 Fáze rozmrazení

Zde je nutné připravit nadcházející změnu. Je nutné informovat vedoucí vývojového oddělení a vedoucí konkrétních týmů, kteří se následně můžou postarat o členy svých týmů. Informovanost vedoucího vývojového oddělení je důležitý, aby si mohl uvědomit kolik času tato změna zabere a jak bude náročná pro jeho týmy. Tedy aby mohl začít s organizováním různých přednášek a školení. Dále je nutné sestavit finanční plán pro majitele, který bude změnu financovat. Připravit zaměstnance (programátory) na nastávající změnu. Součástí rozmrazení je také naplánované časové analýzy.

#### 4.3.5 Fáze provádění změn

Zde bude probíhat samotné provádění změn. Všechny tyto změny budou popsány v následujících kapitolách. Například kapitola 4.6 pojednává o návrhu a integraci nového agilního projektového řízení. Dále kapitola 4.7 následně popisuje návrh a implementaci nového elektronického obchodu pro společnost.

#### 4.3.6 Fáze zmrazení

V této fázi proběhne kontrola všech provedených změn a případně se zde budou řešit problémy, které nastali. Nastane ve chvíli, kdy bude hotový webový obchod pro společnost a všechny týmy budou mít zavedený nový proces projektového řízení a budou se jim všichni řídit. Všechny týmy budou komunikovat pouze se svým **Scrum masterem** a **Product ownerem**, kteří budou fungovat jakožto filtr mezi vedením a programátory, tak i mezi členy týmů. Tyto dvě osoby následně budou komunikovat s vedoucím celého vývoje, který bude zadávat a organizovat úkoly mezi samotné týmy.

### 4.4 Časová analýza

Pokud jde o časovou analýzu, bude zde popsána pouze zjednodušeně v textové podobě. Při návrhu a implementaci této části systému, kde je zohledněna také změna projektového řízení je důležité mít přehled o časovém rozvrhu. Jednotlivé body (počínaje bodem A) začínají poté, co bylo vývojovému oddělení sděleno, že bude nutné inovovat stávající webový obchod společnosti spolu s integrací nového projektového řízení.

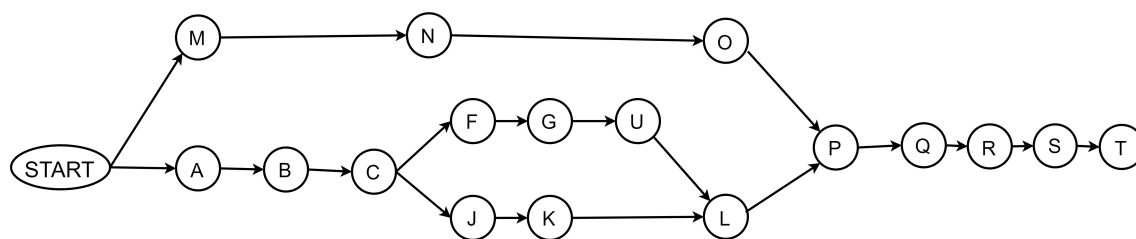
#### 4.4.1 Časové milníky

- **A:** Návrh projektového řízení a konzultace s majitelem společnosti
- **B:** Analýza proveditelnosti a zhodnocení návrhu Agilního řízení
- **C:** Sestavení finančního plánu
- **D:** Schválení plánu ohledně návrhu projektového řízení majitelem společnosti
- **E:** Vedoucí vývojových oddělení informuje vedoucí týmů
- **F:** Naplánování prvotního školení Agile SCRUM externí firmou
- **G:** Účast všech pracovníků na školení
- **U:** Začátek využívání agilní metodiky
- **H:** Výběr scrum mastera a product ownera všech týmů
- **I:** Provedení první ceremonie
- **J:** Zakoupení nástrojů nutných pro nové projektové řízení
- **K:** Konfigurace nástrojů pro agilní vývoj
- **L:** Školení ohledně nástrojů pro agilní vývoj
- **M:** UIX wireframe návrh nového webového obchodu
- **N:** Grafický návrh webového obchodu
- **O:** Schválení tohoto návrhu zadavatelem projektu
- **P:** Grooming ohledně upřesnění technických požadavků mezi členy týmu
- **Q:** Začátek programování
- **R:** Testování
- **S:** Grafické korekce
- **T:** Webový obchod je hotový a projektové řízení aplikováno na tým

#### 4.4.2 Časový graf

Následující obrázek 4.1 zobrazuje veškeré časové milníky, které byly popsány pomocí odrážek výše. Graf vykresluje, jak půjdou jednotlivé události po sobě a co bude nutné dokončit nejdříve.





Obrázek 4.1: Graf časové analýzy (zdroj: vlastní tvorba)

Paralelně budou probíhat dvě časové větve, kde jedna bude moci začít hned a bude se týkat především návrhové části webového obchodu o kterou se postará UIX návrhář a grafická návrhářka, tento návrh by mohl být do měsíce hotový.

Vedle této větve bude probíhat nastavování a různé přednášky/školení o novém projektovém řízení. Přibližný čas těchto školení a přednášek je 1 až 2 měsíce.

Po úspěšném zavedení agilní metodiky SCRUM, budou všichni členové obeznámeni s tím, co mají dělat. Zároveň bude dokončena návrhová část webového obchodu pro společnost. Následně bude započata vývojová etapa. Samotný vývoj by podle odhadů mohl trvat přibližně 2 až 3 měsíce s tím, že by na tomto webovém obchodě pracovali dva programátoři a to přibližně 70 - 80% svého pracovního času.

Ve chvíli, kdy bude produkt (webový obchod) dokončený, alespoň ve fázi, kdy tuto část lze prezentovat a testovat, bude zahájeno testování. Testování bude provádět jiné oddělení, které úzce spolupracuje s vývojem. Jeden člen testovacího oddělení se bude účastnit také stand-upů a schůzek vývojového týmu, aby mohl aktivně sdělovat zpětnou vazbu týmu a mohl s nimi lépe spolupracovat.

Samotné testování, do kterého budou zahrnuté také opravy různých chyb může zabrat přibližně dva týdny. Během této doby bude tato vyvíjená část umístěna na testovací servery, ze kterých lze výsledky prezentovat klientům a majiteli společnosti.

Díky agilní metodice projektového řízení SCRUM budou vývojáři každý den sdělovat ostatním členům týmu svůj postup. Následně jednou za dva týdny, na velké schůzce (ceremonii), budou prezentovat dokončené dílčí části projektu a zároveň obdrží zpětnou vazbu od **product ownera**. Díky této zpětné vazbě a aktivní komunikaci je možné dynamicky měnit během vývoje různé věci. Některé věci mohli být v době návrhu nejasné nebo začaly být postupně v průběhu vývoje klasifikovány jako náročné na realizaci.

## 4.5 Nákladová analýza

Jak již bylo zmiňováno i v předešlých kapitolách, bude společnost potřebovat některé nástroje pro projektové řízení a také pro samotný vývoj programátorů.

### 4.5.1 Školení agile

Externí společnost, která bude provádět několik školení byla doporučena firmě již dříve a jedná se tedy o kolegy z podobného oboru, kteří ale mají v tomto typu projektového řízení již velké zkušenosti. Jednorázová cena celého školení bude **16 500 Kč**. Kupříkladu společnost, zaměřená konkrétně na tyto typy školení, ohledně agilního projektového řízení, POTIFOB, s.r.o., nabízí školení od 10 700 Kč na osobu. Nebo případně e-learningové kurzy od 4 300 Kč na osobu. Zde vybraná společnost nezohledňuje počet školených osob, takže je ve výsledku výhodnější.

### 4.5.2 Technologie a nástroje

Tabulka níže bude detailně popisovat jednotlivé technologie a nástroje a jejich ceny. Jedná se tedy jak o jednorázové ceny za licence u některých nástrojů, tak i o měsíční či roční poplatky.

Nástroj	Cena - pořizovací	Cena - měsíčně
Jira	5 000 Kč	1 600 Kč
Gitlab	2 500 Kč	550 Kč
PhpStorm		450 Kč (na uživatele) (*7)
<b>Celkem</b>	<b>7 500 Kč</b>	<b>5 300 Kč</b>

Tabulka 4.2: Náklady na technologie/nástroje

V některých případech, jako Jira a Gitlab se jedná o jednorázovou licenci, která pokrývá například 100 uživatelů. Což je pro společnost a konkrétně vývojové týmy, popisované v této práci, dostatečné. Další licence, jako pro nástroj PhpStorm, je počítána na jednoho uživatele, je tedy nutné zakoupit licence pro všechny členy týmu (7 členů).

### 4.5.3 Prostory

Společnost využívá své prostory, ale bohužel v limitovaném rozsahu. Pro velké schůzky celého týmu je nutné pronájem větších prostorů spolu s multimediálním zařízením (projektor,

reproduktory, případně mikrofon). Vzhledem k tomu, že tyto prostory budou pronajaty přímo v budově, kde firma sídlí, pouze v jiném patře, nebudou žádné náklady spojené s cestováním. Samotné náklady za prostory jsou vyúčtovány měsíčně a jedná se 1000 Kč za den, kdy je tato větší místnost rezervována firmou. Jak bylo dříve řečeno, společnost (konkrétně tento webový tým) bude provádět tuto velkou schůzi dvakrát do měsíce, tudíž **2 000 Kč za měsíc**.

#### 4.5.4 Vývoj

Vývoj bude rozdělen do dvou částí, kde první je samotný návrh. Do návrhu je započítána práce na 14 dní pro grafického návrháře a UIX návrháře. Grafický návrhář má měsíční plat přibližně 30 000 Kč a UIX návrhář přibližně 40 000 Kč. Jelikož se jedná tedy o práci na dva týdny bude výsledná suma přibližně **35 000 Kč**.

Druhá část nákladů z vývoje budou samotní programátoři. Vývoj zabere přibližně dvě měsíce času pro dva programátory, kde každý má mzdu 45 000 Kč měsíčně. Což za dva měsíce činí dohromady **180 000 Kč** za předpokladu, že tito dva programátoři budou pracovat výhradně na tomto projektu. V časové analýze bylo určeno, že tito programátoři budou pracovat přibližně 80% času na tomto projektu. Zbýlých 20% času je určeno jakožto rezerva různým akutním opravám či řešení jiných prioritních úkolů, které se mohou objevit. Poslední částí vývoje je testování. Testování bude provádět jeden pracovník testovacího oddělení, který bude pracovat spolu s vývojovým oddělením a odhadovaný čas na testování a opravu všeho je přibližně měsíc. Pracovník testovacího oddělení má měsíční plat **28 000 Kč** a bude vyhrazen pouze pro tento projekt.

Mzdy uváděné v této kapitole vychází z průměrných platových ohodnocení pro tyto pracovní pozice v České republice. Zohledněné je taky přibližné platové ohodnocení, jež mají zaměstnanci ve společnosti.

#### 4.5.5 Souhrn nákladů

Do nákladů jednorázových byly zahrnuty pouze licence za některé nástroje, které dohromady činí **7 500 Kč**. Následně cena školení agilního projektového řízení je **16 500 Kč**. Do jednorázových nákladů lze zahrnout také samotný návrh webového obchodu, který byl stanoven na **35 000 Kč** v případě, že bude hotový do 14 dnů.

Pokud jde o samotné měsíční náklady tak do nich lze zahrnout práci programátorů, která byla stanovena měsíční mzdou 45 000 Kč, pro dva programátory to je tedy 90 000 Kč

měsíčně. Následně sem spadají měsíční náklady vycházející z nástrojů nutných pro vývoj (5 300 Kč). Dále pronájem prostorů pro pravidelné schůzky týmu agilního vývoje - **2 000 Kč** za měsíc. Je nutné sem zahrnout také částečnou práci, ať už jenom konzultaci a zpětnou vazbu, ostatních členů agilního týmu, kteří budou svým způsobem pomáhat.

<b>Položka</b>	<b>Cena - jednorázová</b>	<b>Cena - měsíční</b>
Školení	16 500 Kč	
Nástroje	7 500 Kč	5 300 Kč
Návrh	35 000 Kč	
Prostory		2 000 Kč
Programátoři		90 000 Kč
Agilní tým		15 000 Kč
Testování		28 000 Kč
<b>Celkem</b>	<b>59 000 Kč</b>	<b>140 300 Kč</b>

Tabulka 4.3: Souhrn nákladů

V případě optimálního vývoje, kde vše půjde podle představ, zabere vývoj celého webového obchodu včetně návrhu, samotné implementace, testování a samozřejmě i aplikováním agilní metodiky do vývojového oddělení, necelé 4 měsíce. Pořizovací (jednorázové náklady) budou tedy **59 000 Kč**. Měsíční náklady činí sice 140 300 Kč ale ve skutečnosti jsou relativně proměnlivé. První dva měsíce nebude do tohoto vzorce zahrnuto testování. Měsíční suma bez člena testovacího oddělení bude tedy 112 300 Kč. V případě vývoje na dva měsíce se bude jednat o **224 600 Kč**. Poslední měsíc, kdy budou vývoj dokončen a bude probíhat testování budou tedy měsíční výdaje na **140 300 Kč**.

Celkové náklady na přibližně 4 měsíční vývoj budou tedy činit **423 900 Kč** včetně jednorázových investic.

## 4.6 Návrh a zavedení projektového řízení

Součástí vývojové oddělení jsou 3 hlavní týmy. Konkrétně webový tým, jež nejvíce figuruje v této práci je složený z 9 členů, z čehož 7 jsou vývojáři. Tento počet už není natolik nízký, aby šla ignorovat nízká úroveň projektového řízení, a proto byla společnost donucena přijít s návrhem **agilní metodiky SCRUM**. Více informací o agilním přístupu k projektovému řízení v kapitole 2.4.2 a kapitole 2.4.2.

Požadavek na změnu projektového řízení a zavedení nového, přichází od majitele společnosti. Prvotní důvody jsou ty, že majitel společnosti a vedoucí oddělení by měli rádi dříve hotové projekty a celkově by ocenili, kdyby týmy dokázal více efektivně pracovat. Vedoucí vývojového přišel s agilní metodikou SCRUM.

První setkání s touto metodikou přišlo při školení, které bylo vedeno externí společností, která má rozsáhlé zkušenosti s agilním řízením. Na tomto školení bylo všem členům týmů obecně vysvětleno o co vlastně jde při agilním vývoji. Byla zde popsána řada důležitých aspektů pro tento typ projektového řízení. Jednalo se například o sprinty, stand-up, ceremonie, planning. Tohle školení bylo v první části především jako přednáška, druhá část už byla spíše diskuze a názorné příklady.

Další důležitým faktorem bylo stanovení dvou důležitých rolí a to **Product owner** a **Scrum master**. Více informací o těchto rolích je v obecném popisu agilního vývoje SCRUM v kapitole 2.4.2. Na jednu z těchto rolí byla dosazena osoba, která měla nejbližší ke členům týmu a chovala se jako jedna z nich - scrum master. Druhá osoba měla dělat spíše takový filtr a zpracovávat různé zadání projektů a odpovídat na techničtější dotazy - product owner.

#### 4.6.1 Stand-up

Hned po školení bylo prvním úkolem zavedení pravidelných stand-up meetingů. Tyto meetingy jsou důležitým nástrojem, především pro členy vývojového týmu. Existuje několik pravidel, podle kterých by tyto rychlé schůzky **měly probíhat**. Jak již napovídá jejich název, mělo by se jednat o schůzky, kde všichni členové stojí. Tohle pravidlo je zavedené především proto, aby to vytvořilo částečně ne příliš pohodlné a příjemné prostředí a jednotliví účastníci se snažili mluvit co nejvíce stručně, ale zároveň věcně a užitečně. Tohle setkání by mělo ostatním členům týmu objasnit co jeho kolegové aktuálně dělají, na čem pracují, zdali mají ke své práci vše, co potřebují, jestli je něco blokuje a co plánují dělat dále. Každý člen týmu by měl mluvit určité kvantum času, například 3-5 minut a předat sám od sebe slovo dalšímu členovi. Začátky těchto meetingů byly velice krušné, a to především proto, že agilní metodika sama o sobě poskytuje velkou volnost a hodně pravidel je nepsaných a společnosti si je tedy musí přizpůsobit sami sobě. Prvních několik měsíců byly tyto meetingy relativně neefektivní a zabralo dost času je zdokonalit. Pro zlepšení těchto schůzek do fungování týmu byla použita pravidla uvedená v první půlce tohoto odstavce. Dalším zajímavým pravidlem, které se efektivně osvědčilo, bylo tyto schůzky

provádět těsně před pauzou na oběd. Členové týmu se museli držet pravidel, říct věci co dělají/dělali, co dělat budou a co je blokuje. Zároveň měli hlad a už se těšili na nadcházející pauzu, takže se tyto schůzky poněkud zrychlily. Další zajímavou pomůckou bylo například pravidelné přání dobré chuti ve francouzštině od scrum mastera - *bon appétit*, na konci těchto schůzek.

#### 4.6.2 Ceremonie

Na každou druhou středu byla naplánována ceremonie. Jedná se o klasický meeting složený ze tří částí: *demo*, *retrospektiva* a *plánování*. Prvních několik schůzek bylo na téměř celý den. Což bylo velice psychicky náročné, vyčerpávající a ve skutečnosti i neefektivní. Některé tyto schůzky zakončovaly také různé hádky mezi členy týmu. První problémy zde byly především stanovení pravidel. Jak již bylo dříve zmíněno, agilní řízení má poněkud volnou strukturu a je na společnosti, aby si určité věci přizpůsobila svým potřebám. Na jednu stranu někteří členové chtěli využít striktní pravidla jiných firem a někteří členové si chtěli najít svá vlastní a své vlastní způsoby a zkusit je.

První část tohoto meetingu je tedy **demo**. Tato část na některých schůzkách bývá přeskočena nebo bývá velice krátká. Jednotliví členové týmu zde mají prostor pro prezentování nových částí systému, na kterých pracovali nebo které již dokončili. Samozřejmě je to především na nich, pokud některý člen nemá žádné viditelné výsledky a pracoval například na logice některé části systému, tak samozřejmě ničemu nevadilo, že neměl co prezentovat. Další částí je **retrospektiva**. První návrh v této části byl takový, že členové týmu budou poskytovat zpětnou vazbu Scrum masterovi a Product ownerovi. Po této zpětné vazbě se očekávalo to stejné i od product ownera a scrum mastera. Další verze byla taková, že členové týmu tuto retrospektivu dělali bez dozoru scrum mastera a product ownera a poté jim sdělili své výsledky, ani tahle verze nebyla příliš užitečná. Jednotlivé poznatky z této retrospektivy začaly být postupně nekonstruktivní, neužitečné a opakovaly se. Tato část ceremonie by měla správně sloužit ke zlepšování věcí v týmu či případně v celkovém agilním vývoji. Finální verze retrospektivy je tedy taková, že scrum master by měl částečně řídit a také řídí tuto retrospektivu, může ji různě inovovat tím, že přijde s novou formou. Může například využít brainstorming, různé typy canbanů apod. Účelem tedy je konstruktivní zpětná vazba, co šlo v daném sprintu dobře, co šlo špatně, jak pro příště vylepšit nebo eliminovat ty věci, které šly špatně.

Poslední části a pravděpodobně časově nejrozsáhlejší je **plánování**. Tato část byla svým způsobem relativně jasná, alespoň co se týče pravidel pro plánování úloh, na nichž se bude v následujícím sprintu pracovat nebo které mají být v daném sprintu úspěšně dokončeny. Více informací o sprintu samotném v následující kapitole 4.6.3. Účelem této části bylo tedy stanovení obsahu sprintu (úkoly), stanovení jednotlivých bodů pro každý úkol, určení cíle sprintu a odstartování. Tímto byla ceremonie zakončena.

### 4.6.3 Sprint

Samotný sprint je poněkud rozsáhlá věc. Jako u všech věcí je i zde agilní vývoj poněkud benevolentní v určování pravidel. Základem sprintu je tedy stanovení úloh, které budou součástí sprintu. Tyto úlohy určuje product owner. Komunikuje s řadou dalších oddělení, s majitelem firmy či případně s partnery a zjišťuje jaké jsou konkrétní požadavky a ty následně formuluje do zadání.

Toto zadání úlohy musí být kompletní a musí obsahovat všechny důležité části nezbytné pro vývoj. Například pro vývoj zde zmíněné části informačního systému - webového obchodu byla určena řada samostatných úkolů, každý tento úkol musel obsahovat **UIX**<sup>2</sup> návrh, přesně definovaný popis úlohy a také jak má být daná úloha otestována týmem testerů. V poslední řadě měl být určen počet **story points**.

Tyto body nastavuje tým jako celek. Počet těchto bodů by měl vyjadřovat obecnou náročnost úkolu. Samozřejmě zde byla diskuze o tom, že některý úkol zabere jiné časové kvantum pro senior programátora a jiné pro junior programátora. Tudíž tento počet bodů nesmí odrážet žádný časový údaj, měla by to být tedy objektivní hodnota náročnosti úkolu. Hodnoty bodů byly stanoveny na násobky čísla 2, takže například 2, 4, 8, 16. V případě, že úkol byl úspěšně dokončen před konce sprintu, byly body za tento úkol přičteny k celkovým doručeným bodům za sprint. Zde se následně vyskytl problém ohledně nedokončení úkolů. V případě, že některý pracovník nedokončil úkol, na kterém pracoval a sprint byl ukončen, byl tento úkol přesunut do následujícího sprintu. Samozřejmě nedoručení těchto bodů bylo viditelné ve výsledném grafu sprintů, kde se porovnává počet bodů za všechny úkoly, jež jsou do sprintu přidány, vůči bodům, které jsou na konci doručeny.

Další důležitou částí sprintu je jeho **cíl** (sprint goal). Čeho chce tým dosáhnout po dokončení daného sprintu? Cíl sprintu by měl být dosažitelný pomocí splnění jednoho nebo více úkolů ve sprintu, ne nutně všech.

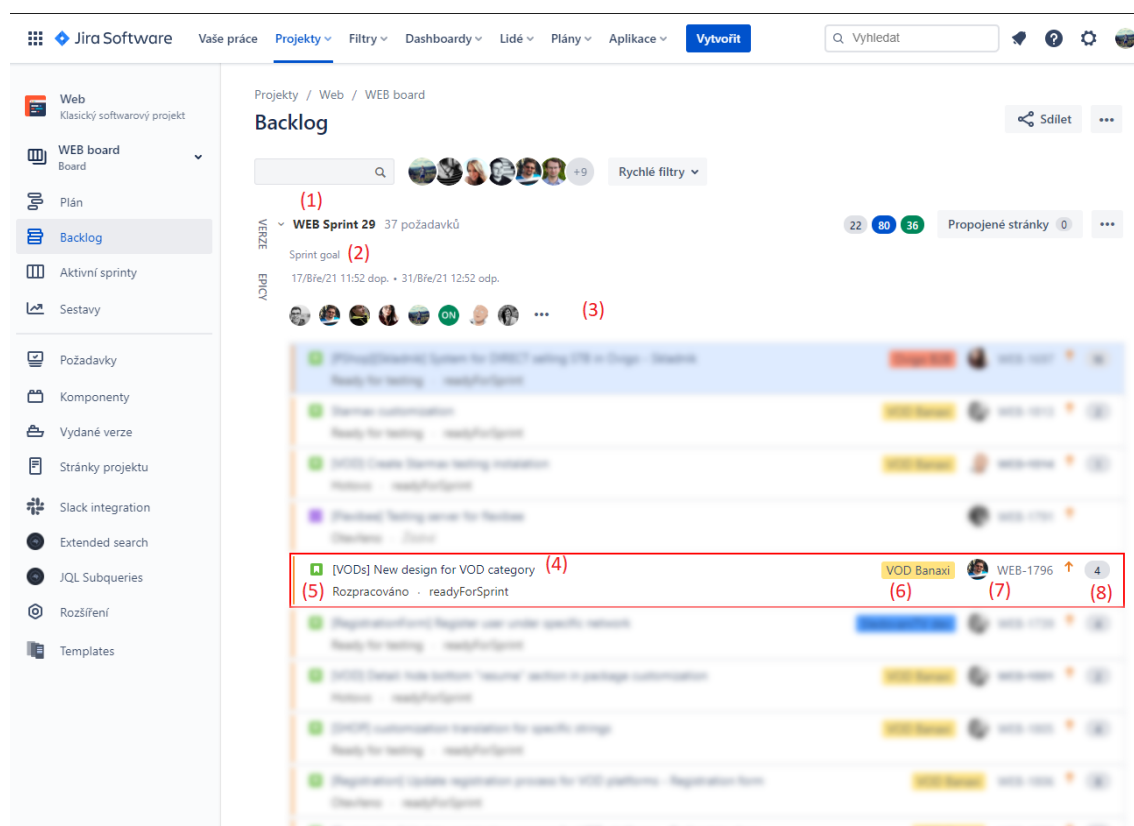
---

<sup>2</sup>UIX - řada technik využitých pro vývoj uživatelského rozhraní včetně grafického návrhu

Po zahájení sprintu si jednotliví členové týmu rozeberou úkoly podle své vlastní úvahy. Úkoly zde tedy nejsou nijak striktně přiřazovány a je tedy na tom, jak se tým rozhodne, kdo co bude dělat.

#### 4.6.4 Nástroj Jira

Zde zmiňovaný nástroj **Jira** slouží k samotné organizace práce. Plánování celého sprintu a všech jeho úkolů se děje právě zde. Tento on-line nástroj dokáže vizualizovat průběh celého sprintu a všech jeho úkolů, viz obrázek 4.2. Díky novému agilnímu projektovému řízení je tento nástroj integrován do vývojového týmu a bude nutné jej využívat.



Obrázek 4.2: Jira - ukázka sprintu (zdroj: vlastní tvorba)

Obecný popis sprintu byl vysvětlen v kapitole 4.6.3, v obrázku výše je tento sprint vizuálně zpracován. Jsou zde viditelné všechny důležité body.

1. označení současného sprintu (níže je vidět i od kdy do kdy probíhá)
2. cíl sprintu - viz kapitola 4.6.3 (čeho má být v daných dvou týdnech dosaženo)
3. filtr - lidé jež aktuálně pracují na sprintu (testéři + programátoři + zadavatelé)

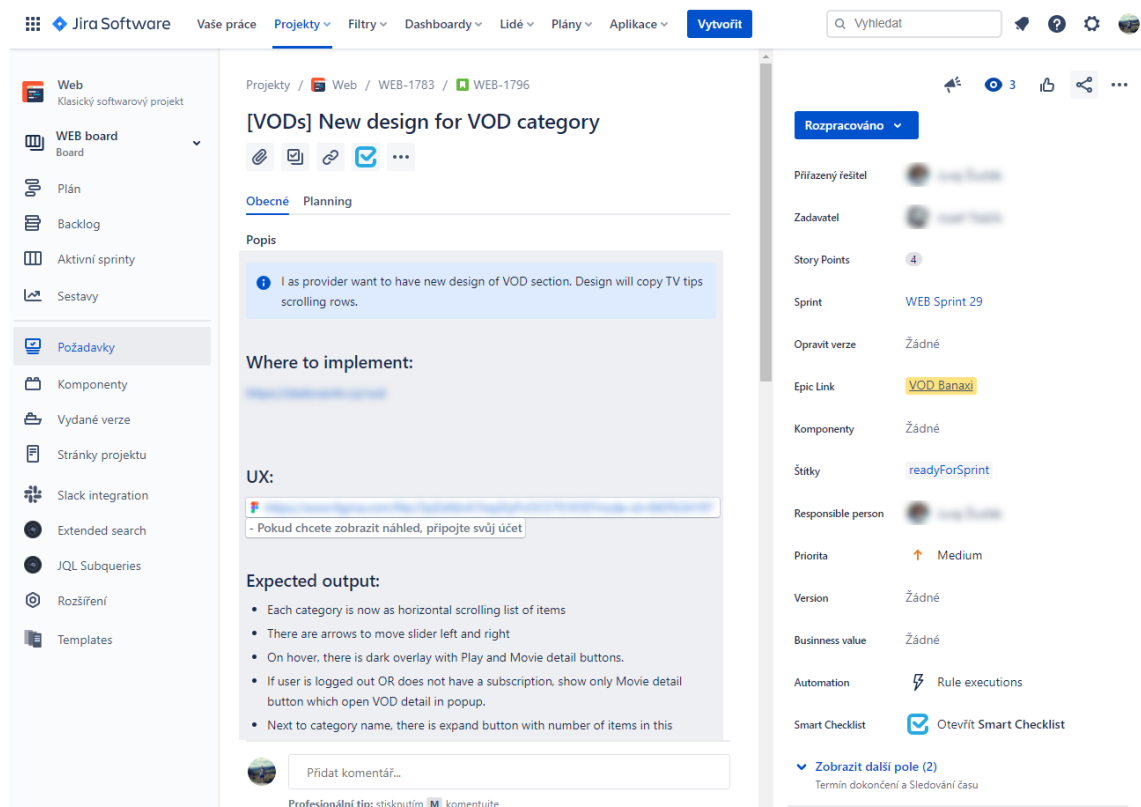


4. název úkolu, hranatá závorka určuje sekci, do které úkol zasahuje
5. aktuální stav úkolů (otevřeno, rozpracováno, připravené na testování, zrušeno, hotovo)
6. pro koho je úkol určený, společnost, partner či případně sekce
7. přiřazená osoba, která aktuálně na daném úkolu pracuje (programátor/tester)
8. počet story points

Mimo samotného sprintu je v tomto nástroji velice dobře graficky zpracován samotný **úkol**. Nabízí velkou škálu nastavení, které mohou sloužit například pro přiřazování zodpovědných osob, různých štítků, stavů či případně propojovat jednotlivé úkoly spolu navzájem. Dále je možné úkoly různě komentovat. Tento *chat* pod úkolem je tedy určen pro otázky od programátorů k zadavateli úkolu, který jim následně odpoví právě k těmto komentářům. Chat je zde důležitý především proto, pokud má někdo doplňující otázky, které mohou být důležité pro vypracování, je nutné je sdělit viditelně, aby kdokoli mohl následné odpovědi využít pro správně dokončení úkolu. Ukázka úkolu je na obrázku [4.3](#).

Samotné propojování úkolů může mít několik typů. Pokud k nějakému úkolu připojím jiný úkol pomocí propojení "**je blokován**" tak aktuální úkol bude ve stavu "Blokován" a bude čekat na vyřešení propojení úkolu.

Dalším typem je propojení pomocí vazby "Related" to znamená že dva dané úkoly se týkají například stejné sekce nebo mají sdílenou funkcionalitu.



Obrázek 4.3: Jira - ukázka úkolu (zdroj: vlastní tvorba)

#### 4.6.5 GitLab

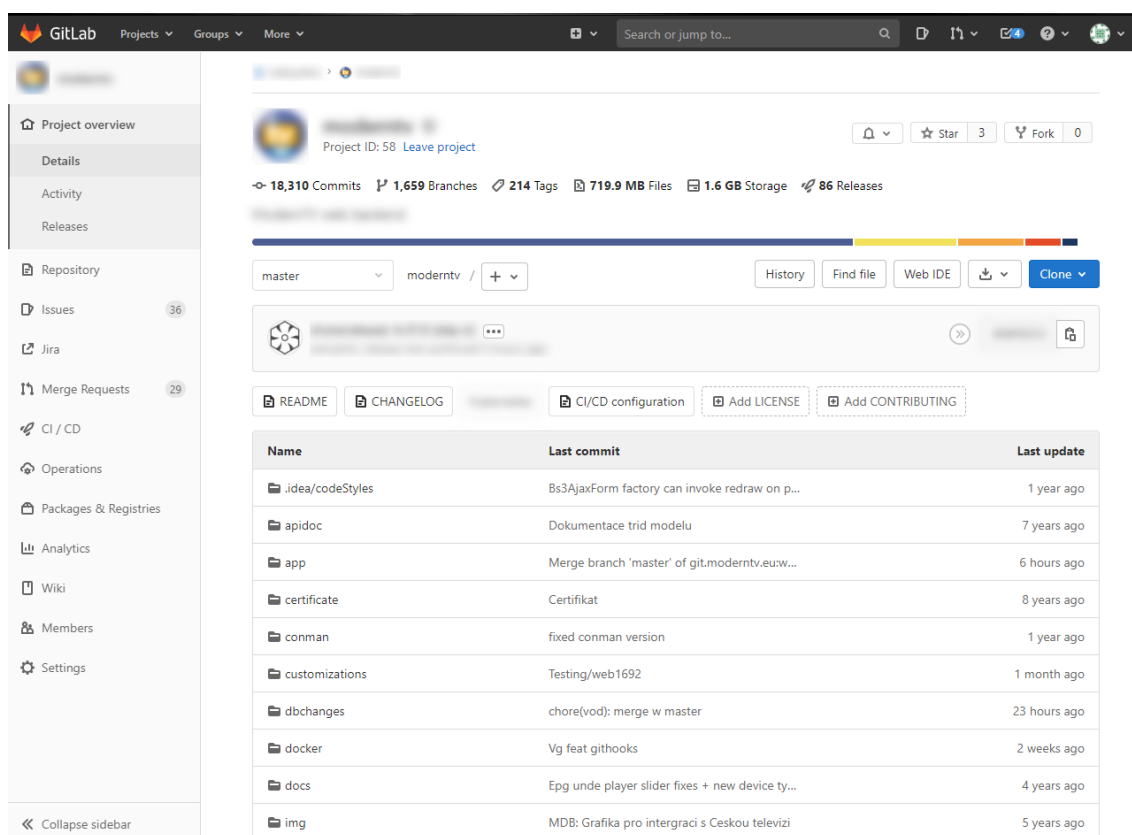
V obecném měřítku je tento nástroj popsán v teoretické části práci v kapitole 2.3.7 a 2.3.7, zabývající se systémem git a nástrojem gitlab. Společnost a členové vývojového oddělení a konkrétně webového týmu, tento nástroj využívají pro různé věci. Hlavní z nich je nahrávání částí nebo celku vypracovaných úkolů na webový repozitář<sup>3</sup>, kde si následně mohou jednotlivé verze porovnávat s produkční verzí<sup>4</sup>. Následně při porovnávání svých změn na projektu mohou programátoři vytvořit tzv. *merge request*, který je blíže popsán v následující kapitole 4.6.5. Dále je součástí tohoto nástroje také Wiki<sup>5</sup>.

Pro lepší představu jak samotný nástroj vypadá je zde obrázek 4.4. Na obrázku je vidět několik důležitých věcí. Seznam souborů uprostřed stránky, hned nad nimi je vodorovná čára, znázorňující poměry různých jazyků zdrojových kódů. Dalším důležitým článkem je zde levé menu ve kterém je několik klíčových odkazů: Jira, Wiki, různé analytické nástroje a především **Merge requesty**.

<sup>3</sup>Webový repozitář - datové úložiště pro zdrojové kódy a různé soubory

<sup>4</sup>Produkční verze - aktuální webová verze projektu, stabilní, otestovaná a funkční

<sup>5</sup>Wiki - obsahuje editor pro tvorbu dokumentace daného projektu, je propojená s repozitářem

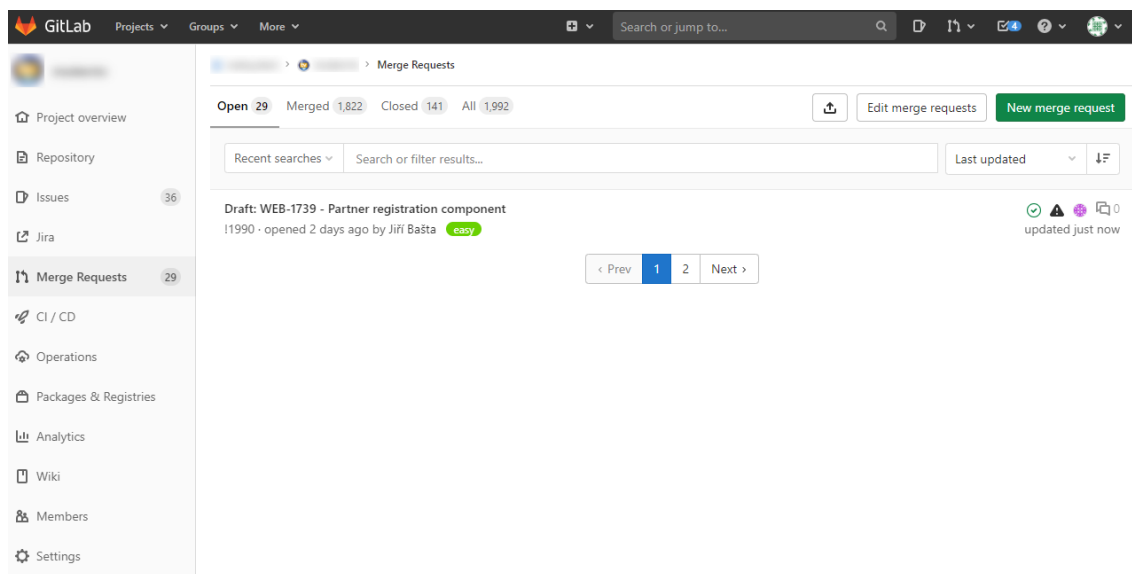


Obrázek 4.4: GitLab - ukázka projektu (zdroj: vlastní tvorba)

## Merge request

Pokud někdo ze členů týmu pracuje na svém úkolu, musí se řídit v současné době několika kroky. Mezi tyto kroky patří vytvoření větve (**branch**) viz kapitola 2.3.7. Název této větve vychází z identifikačního čísla úkolu, vygenerovaným v nástroji Jira. Tato větev následně slouží k nahrávání všech změn, které programátor udělá. Současně může vytvořit merge request, ve kterém jsou tyto změny vidět. Všechny změny jsou porovnávány s aktuální verzí produkčních zdrojových kódů. Nahrávání na tento externí repozitář je zároveň také bezpečnostní pojistkou, pokud by vývojáři vypověděl počítač nebo jeho lokální úložiště (disk).

Z názvu **merge request** lze vyčíst, že je to určitá žádost k tomu, aby data vypracovaného úkolu byla sloučena a nahrána na produkční verzi projektu. Tento úkon provádí jeden z nadřazených programátorů, který má dostatečnou znalost celého projektu a je tedy schopen provést code review (více informací v další kapitole). Seznam těchto žádostí je vidět na obrázku 4.5.



Obrázek 4.5: GitLab - ukázka seznamu requestů

Jak již bylo řečeno, každý merge request je tedy určitým způsobem spojený s daným úkolem. Při prvotní inicializaci toho requestu je použito označení **WIP**<sup>6</sup> nebo **Draft**<sup>7</sup>.

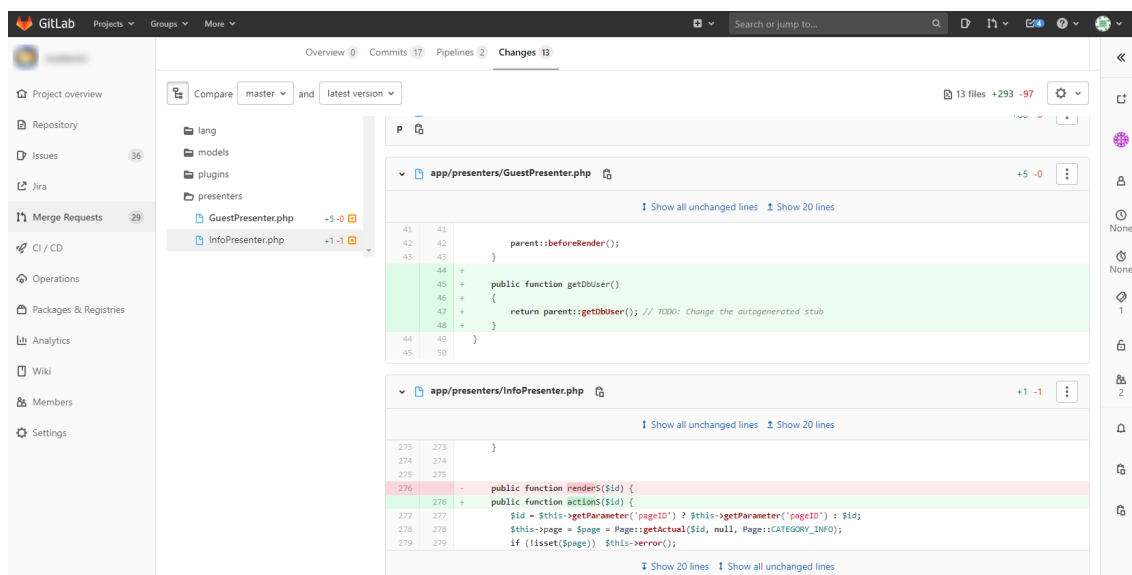
---

<sup>6</sup>WIP - z angl. Work In Progress - rozpracovaný

<sup>7</sup>Draft - pracovní verze nebo náčrt

## Code review

Je prováděno zkušenějším a většinou nadřazenějším programátorem, velice často vedoucím týmu. Tento programátor by měl mít znalosti téměř celého systému a také daného jazyka a měl by tedy dokázat odhalit případné chyby. Následně by měl provést také analýzu z pohledu optimalizací a určitých standardů, které vývojový tým dodržuje. Toto hodnocení kódu probíhá v samotné žádosti (viz obrázek 4.6), kde jsou vidět všechny změny, které programátor provedl. Každá z těchto změn může být různě okomentována a může vývojář, který změnu provedl, může být požádán o vysvětlení nebo o úpravu. Ve většině případů je tento programátor také ten dotýčný, který nahrává úpravy na produkční verzi a poté sleduje různé výkyvy či případné výpadky a zpomalení služeb. Pokud nahrávání změn způsobí nějaké chyby či výpadky služby, může být následně systém uveden do původního stavu během několika sekund/minut.



Obrázek 4.6: GitLab - ukázka seznamu requestů (zdroj: vlastní tvorba)

## Testování

Ve chvíli, kdy si programátor vytvoří novou větev pomocí konvencí jež byly popsány v kapitole 4.6.5 a následně do této větve nahraje svoje změny a odešle je na vzdálený repo-zitář, je vytvořen testovací server. Tento testovací server je vytvářet automatizovaně. Je nastavena doména například *web1234.testing.spolecnost.cz* na kterou má následně kdokoli přístup. Na této adrese je přístupný celý projekt (produkt) včetně nových změn, které programátor vytvořil. Tento testovací web je ve skutečnosti kopií produkčního webu s tím,

že má vlastní databázová data. Po dokončení úkolu ze strany programátora je následně odeslán odkaz na tento testovací web a člen testovacího týmu může otestovat nové nově implementované části. Moje vlastní zkušenost a můj názor na tyto testovací servery je takový, že tyto servery mají jeden zásadní nedostatek. Samozřejmě by se jednalo o relativně velkou shodu náhod, ale vzhledem k tomu, že na testovací servery programátor průběžně může nahrávat rozpracovaný stav svého úkolu, tak se tam může vyskytnout řada chyb. Tyto chyby mohou případným útočníkům poskytnout velice jednoduchou možnost, jak si zajistit některá citlivá data, vzhledem k tomu, že je zde využita kopie produkční databáze. V tomto ohledu je zde mnoho možností a prostoru pro zlepšování.

## 4.7 Návrh a zavedení webového obchodu

Další a velice důležitou částí této práce je samotný návrh a implementace dílčí části informačního systému - konkrétně webového obchodu (tzv. e-shop). Informace a grafické náhledy staré verze tohoto obchodu jsou v kapitole 3.2.7. Důvody pro inovaci, respektive pro návrh a implementaci nového obchodu jsou obsaženy v analýzách firemního prostředí v kapitole 3.3 a samotné výsledky a zhodnocení všech analýz v kapitole 3.3.5.

Je důležité si uvědomit primární důvod, proč je tato práce složena ze dvou poměrně rozsáhlých částí, kde první část se zabývá projektovým řízením a druhá samotným návrhem. Hlavním důvodem tedy je, jak nové projektové řízení dokáže ovlivnit implementaci takto rozsáhlé části systému. Po provedených analýzách lze říci, že bez úpravy projektového řízení a zavedení agilní metodiky by tato změna trvala několikanásobně více času nebo by nebyla realizovatelná.

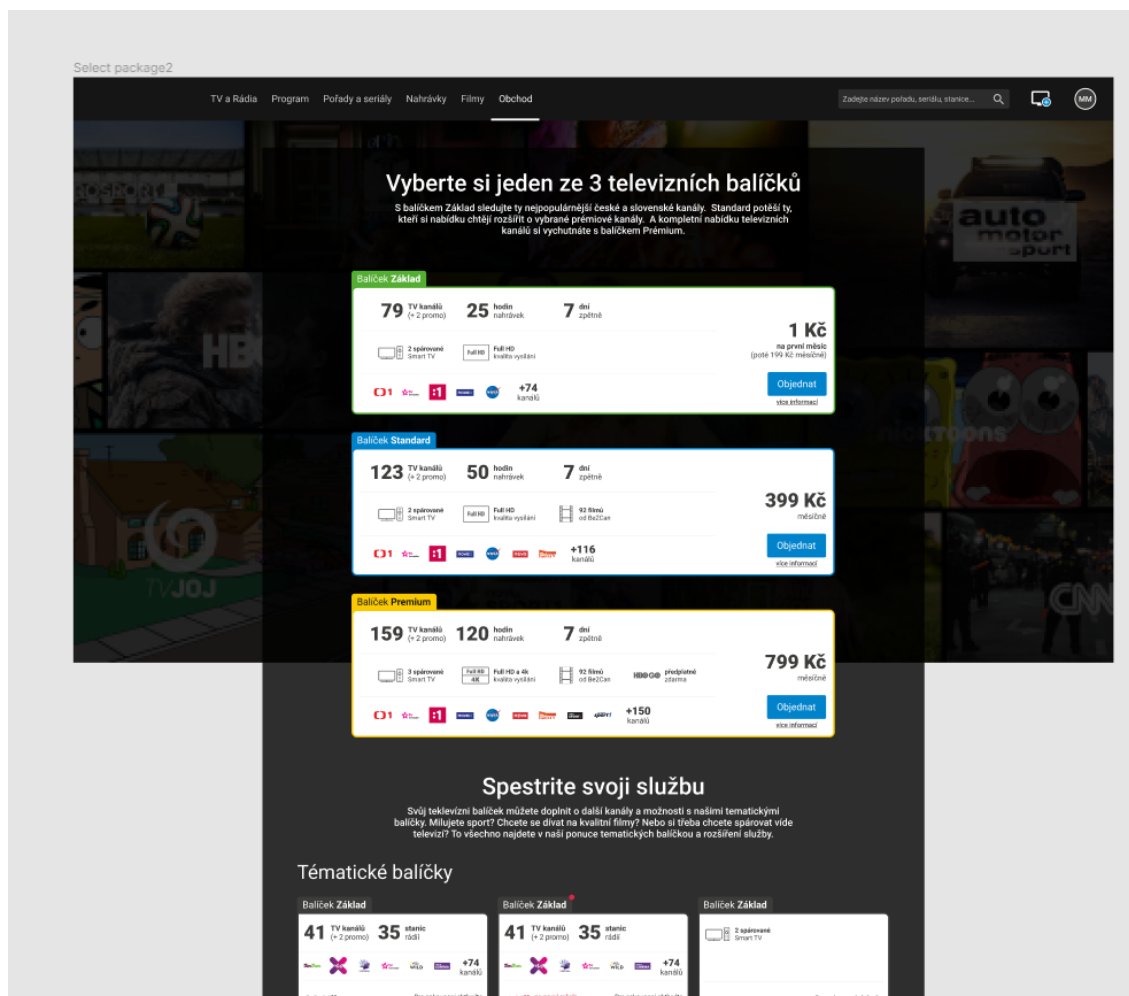
### 4.7.1 Návrhová část

#### Wireframe

Wireframe neboli drátěný model je využíván především v navrhování webových stránek. Slouží k definování rozložení jednotlivých prvků a jejich samotné funkcionalitě. Jde tedy spíše o takový logický návrh, kde nemusí být poskytnuty žádné grafické návrhy. Je tedy teoreticky možné započít práci na logické funkcionalitě úkolu jenom díky tomuto modelu.

#### Grafický návrh

Jak samotný název vypovídá, jedná se už tedy o graficky zpracovaný návrh úkolu. Tento návrh se provádí pomocí nástroje **Figma**. Tento nástroj je vhodný jak pro drátěné modely, tak i pro samotné grafické zpracování. Jedná se o profesionální nástroj, který grafikům nabízí velice pestrou škálu návrhářských možností. Samotný grafický návrh lze pro lepší představu vidět na obrázku 4.7.



Obrázek 4.7: Grafický návrh webového obchodu (zdroj: vlastní tvorba)

## Riziková politika

V této kapitole proběhne analýza rizik a jejich ohodnocení. Průběh, vyhodnocení a celková vizualizace rizik bude probíhat pomocí mapy rizik. V následující tabulce je zobrazený seznam všech rizik, které byly definovaný po brainstormingu některých členů týmů a jejich vedoucích. Jednotlivá rizika byla ohodnocena pomocí skórovací metody. Hodnocení probíhá tedy v těchto hodnotách:

## Pravděpodobnost

- téměř žádná ( $1 - 2 = 0\%$  až  $19\%$ )
- nízká ( $3 - 4 = 20\%$  až  $39\%$ )
- pravděpodobná ( $5 - 6 = 40\%$  až  $59\%$ )
- více pravděpodobná ( $7 - 8 = 60\%$  -  $79\%$ )



ID	riziko	výskyt (pst)	dopad	skóre
R1	Změna požadavků majitele	2	5	10
R2	Neschválení plánu majitelem	1	9	9
R3	Nesouhlas (nízká spolupráce) zaměstnanců	6	5	30
R4	Špatné organizační schopnosti (zpomalení)	4	6	24
R5	Externí společnost nedodrží termíny	3	3	9
R6	Odchod klíčových zaměstnanců	2	8	16
R7	Výpadek internetového připojení ve firmě	2	4	8
R8	Překročení finančního rozpočtu	8	3	24
R9	Nízké zabezpečení ve společnosti (útok)	2	6	12
R10	Nesplnění časového plánu	7	5	35

Tabulka 4.4: Tabulka rizika jejich ohodnocení (zdroj: vlastní tvorba)

- vysoká pravděpodobnost (9 - 10 = 80% - 100%)

### Dopad

- minimální (1 - 2)
- méně významný (3 - 4)
- významný (5 - 6)
- velmi významný (7 - 8)
- kritický (9 - 10)

**Hodnota rizika** pravděpodobnost x (krát) dopad

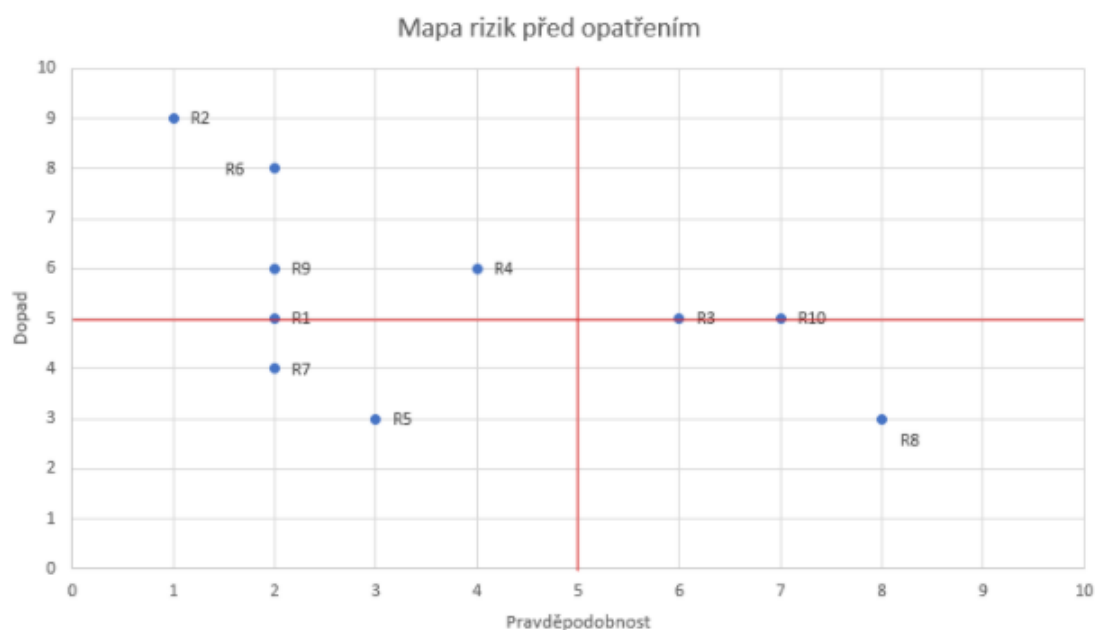
Tabulka 4.4 výše popisuje rizika, které mohou nastat při realizaci plánované změny. Jednotlivá rizika jsou hodnocena pomocí pravděpodobnosti výskytu, míra dopadu a výsledné skóre, který je součinem předchozích dvou sloupců.

Tabulka níže (??) popisuje rozdělení samotné mapy rizik do čtyřech kvadrantů, kde každý kvadrant odpovídá jiné klasifikaci rizika.

Graf níže (??) popisuje vizuální formou stejná rizika, jako byly výše v tabulce. Tyto rizika jsou zanesena do grafu o 4 oddílech, kde každý kvadrant má velikost 5x5.

<b>významná rizika</b>	<b>kritická rizika</b>
<b>bezvýznamná rizika</b>	<b>běžná rizika</b>

Tabulka 4.5: Rozdělení mapy rizik (zdroj: vlastní tvorba)



Obrázek 4.8: Mapa rizik (zdroj: vlastní tvorba)

Z grafu výše je patrné která rizika spadají do kterého kvadrantu. Rizika v horní části grafu jsou ty nejdůležitější (významná a kritická) a je tedy nutné se jimi zabývat. Jedná se tedy o rizika **R2**, **R6**, **R9**, **R4** a následující tři rizika, která jsou na pomezí spodní a horní části grafu **R1**, **R3** a **R10**. Pro rizika uvedená výše, v kritické sekci, budou navržena protiopatření.

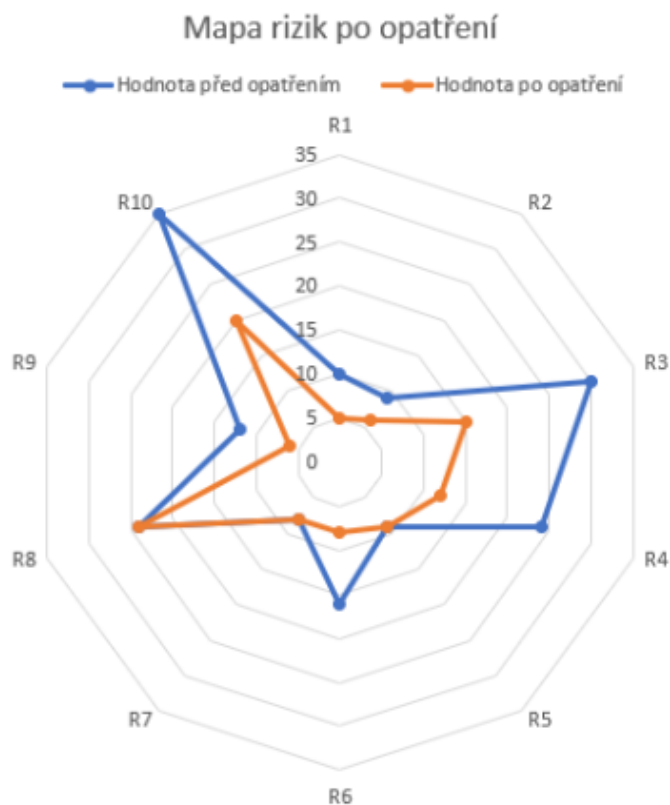
Zbývá rizika R5, R7 a R8 jsou označena jako bezvýznamná nebo běžná rizika, a tedy není úplně prioritní se jimi zabývat.

Zde budou řešena především kritická a významná rizika vyskytující se v horní polovině mapy rizik. V tabulce níže (4.6) jsou tedy uvedena přesně tato rizika a pro ně budou provedena následující protiopatření pro snížení jejich dopadu.

ID	Riziko	Protiopatření	Nový výskyt	Nový dopad	Nové skóre
R1	Změna požadavků majitele	Detailní písemné ujasnění požadavků	1	5	5
R2	Neschválení plánu majitelem	Alternativní plán s méně náklady	1	6	6
R3	Nesouhlas (nízká spolupráce) zaměstnanců	Bonusy, lepší motivace zaměstnanců	3	5	15
R4	Špatné organizační schopnosti (zpomalení)	Detailní vypracování plánu nebo přenechání externí společnosti	2	6	12
R6	Odchod klíčových zaměstnanců	Proškolení všech zaměstnanců pro případné nahrazení	2	4	8
R9	Nízké zabezpečení ve společnosti (útok)	Záloha dat, rozdělení na více serverů, lepší firewall	2	3	6
R10	Nesplnění časového plánu	Pravidelné kontroly průběhu	4	5	20

Tabulka 4.6: Tabulka rizik - opatření (zdroj: vlastní tvorba)

Graf níže (obrázek 4.9) zobrazuje mapu rizik po zavedení opatření. Opatření byla provedena pro významná a kritická rizika, pro zbylé zůstává hodnota nezměněna. Modrý graf znázorňuje mapu před opatřením a oranžový naopak po zavedení opatření.

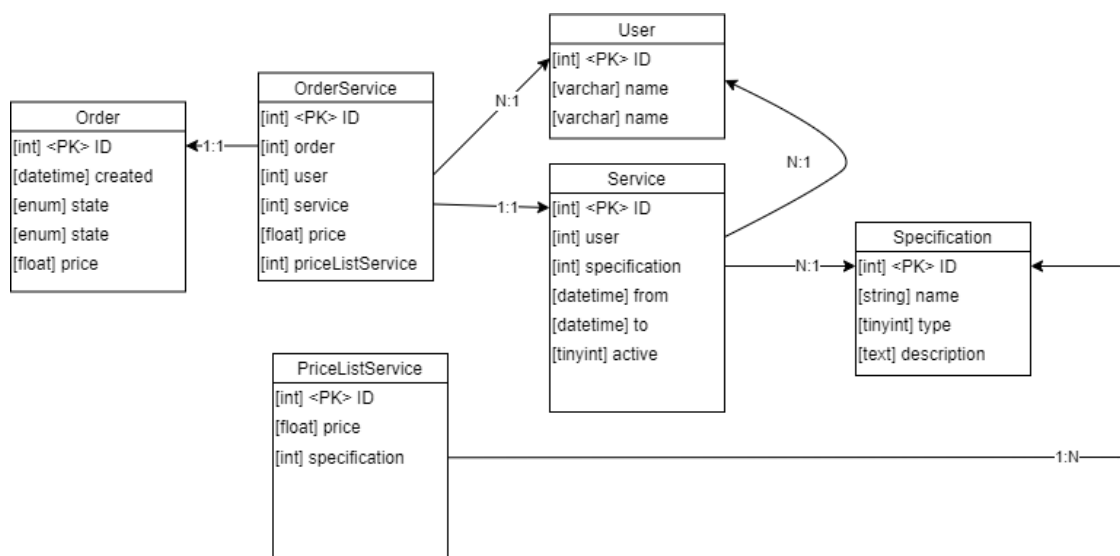


Obrázek 4.9: Mapa rizik po opatření (zdroj: vlastní tvorba)

Díky této analýze následně určit řadu rizik a jejich případná protiopatření. Je vidět, že efektivní projektové řízení zde bude velice důležité. Stejně tak správně formulované zadání a komunikace mezi týmy a samozřejmě i zásah zainteresovaných stran. Obecně lze říci, že díky novému agilnímu projektovému řízení by návrh a implementace takto rozsáhlé části informačního systému **mohl být realizovatelný**.

### ERD diagram

Následující obrázek 4.10 zobrazuje návrh části relační databáze, která bude sloužit pro uchovávání objednávek. Z důvodů cenzury některých firemních dat a struktur je návrh pojat velice zjednodušeně a jsou zde vykresleny jenom některé základní vztahy a základní sloupce tabulek. Detailní popis jak jsou všechny tyto tabulky využity je v samotné realizaci obchodu a konkrétně v procesu objednávky v kapitole 4.7.2.



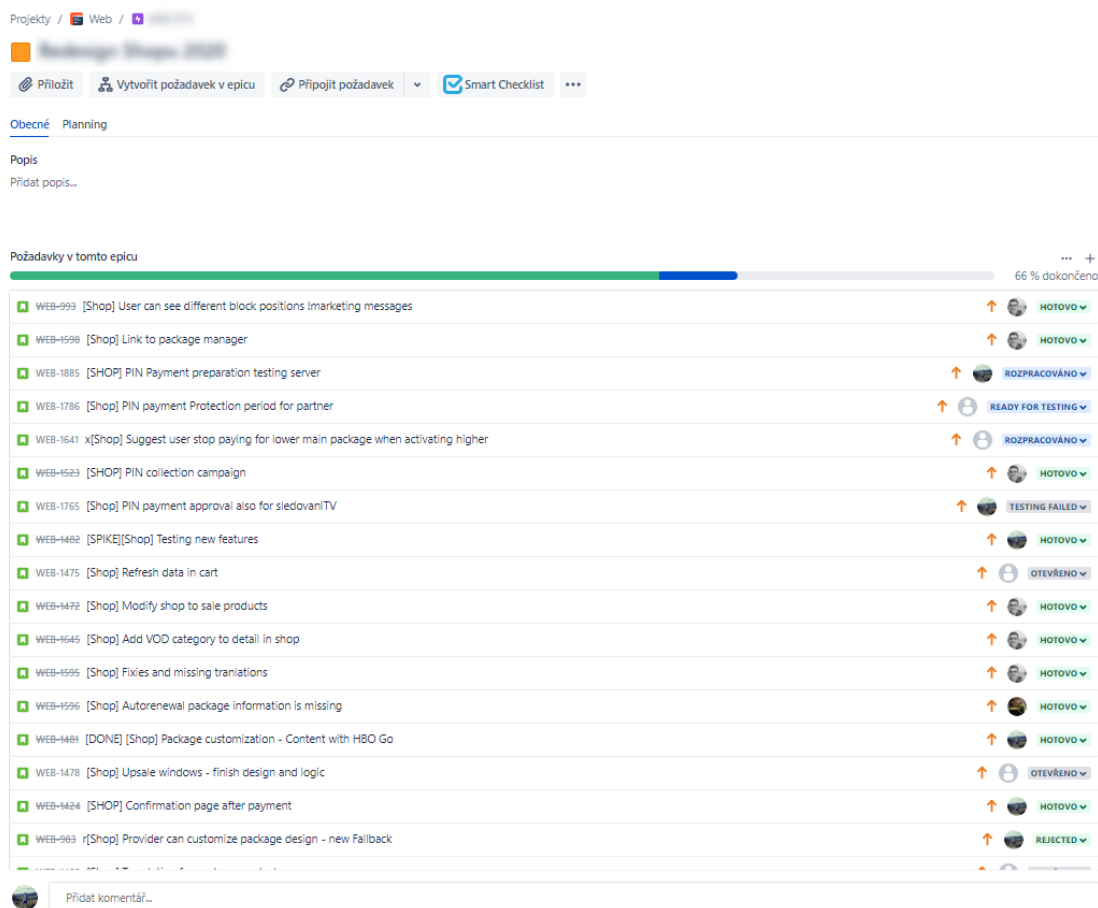
Obrázek 4.10: ERD diagram objednávek (zdroj: vlastní tvorba)

## Jira - úkoly

Jak již bylo zmíněno v dřívějších kapitolách (například v kapitole 4.6), bude tato část informačního systému vyvíjena za pomoci agilního projektového řízení SCRUM. Samotný nástroj Jira je popsán v kapitole 4.6.4. Pro návrh a implementaci tohoto úkolu byla využita řada sprintů. Byl vytvořen tzv. **epic**<sup>8</sup>. Tento epic obsahuje všechny úkoly, které je nutné dokončit k úspěšnému vyhotovení finálního produktu, konkrétně nového webového obchodu. Příklad úkolů ať už hotových nebo rozpracovaných je na obrázku 4.11.

Tyto jednotlivé úkoly mají různé vzájemné závislosti, některé mají určité priority a blokují další úkoly. Pracovníci si tedy vybírají dle priorit úkoly, které mají být dokončeny dříve. Vývoj nového webového obchodu je tedy realizovatelný především díky novému projektovému řízení, které přispívá především lepší a efektivnější organizací práce a rozdělování úkolů.

<sup>8</sup>**Epic** - využitý v agilní metodice SCRUM, slouží k seskupení několika úkolů nebo kombinací úkolů, rozsah může být několik sprintů a výstupem je určitý produkt



Obrázek 4.11: Jira sprint - Epic webového obchodu (zdroj: vlastní tvorba)

## 4.7.2 Realizace

Zde bude popsána samotná realizace a zavedení dílčí části informačního systému - webového obchodu.

### Moderní design a přizpůsobení

Jedním z prvků nového obchodu je nový moderní design (obrázek 4.7), jehož základem jsou tmavší barvy. Obrázek na pozadí s tmavým překrytím bude v budoucnu využitý i na dalších stránkách systému, které budou postupně inovovány a převáděny do tmavého designu. Samotný vzhled této hlavní stránky obchodu lze do určité míry modifikovat. Jednou z těchto modifikací je výše zmíněné pozadí. Jak bylo zmíněno dříve v kapitole 3.2.4, existuje zde nastavení tohoto webového obchodu, obrázek 4.12.

**Obchod**

Try package specification	Žádné	■
Design	New design	■
Banner 1		■
Banner 2		■
Banner 3		■
Banner 4		■
Recommend packages?	Ne	■
Public shop shown like for normal users	Ne	■

**SHOP**

☐ Hlavní balíčky nahore ■

Obrázek 4.12: Přizpůsobení obchodu (zdroj: vlastní tvorba)

V tomto přizpůsobení půjde nakonfigurovat různé prvky obchodu. Prvním je odkaz na zkušební balíček. Zkušební balíček je speciální verzí balíčků, které nejsou zpoplatněny a jedná se spíše o takový dárkový (zkušební) balíček, který klientovi zpřístupní některé základní programy na určitou dobu (například 14 dní). Každá customizace (například každý partner) může mít rozdílné zkušební balíčky. Volba tohoto balíčku je zde nutná kvůli podmínkám, pro zobrazování různých reklamních ploch. Kupříkladu, pokud měl uživatel někdy v minulosti pouze zkušební balíček nebo případně některý z dodatkových balíčků, tak je vhodné mu doporučit některé hlavní balíčky, které jsou výhodnější a poskytují větší nabídku služeb. Další příklad může být uživatel, který si v minulosti opětovně zakoupil balíčky, které mu ale v současnosti již vypršely, proto mu budou zobrazovány reklamy těchto balíčků, aby ho ponoukali k jejich opětovnému zakoupení. Jedná se spíše o takové reklamy, které jsou pro každého uživatele jiné. Položka *"Recommended packages"*- *"doporučené balíčky"* aktivuje nebo deaktivuje tyto reklamy balíčků na míru pro uživatele.

Další možností je přepínání nového a starého vzhledu webového obchodu. Existují například partneři, kteří si prozatím nebudou přát využívat nový vzhled, a tudíž je zde tohle nastavení nutné. Díky tomu, že v projektu bude zachován také starý obchod, je možné využít také AB testování<sup>9</sup>.

Následující čtyři možnosti se týkají zobrazování reklamních proužků (tzv. *banner*). Konkrétně tyto reklamy vychází ze statických stránek (viz kapitola 3.2.5) a do těchto polí se zadávají identifikační čísla dříve zmíněných stránek. Je možné do jednoho pole zadat

<sup>9</sup> **AB testování** - tato metoda porovnává výkon/využití určitých dvou totožných částí produktu, kde jedna je originální beze změny a druhá je ta inovovaná na kterou byla aplikována změna

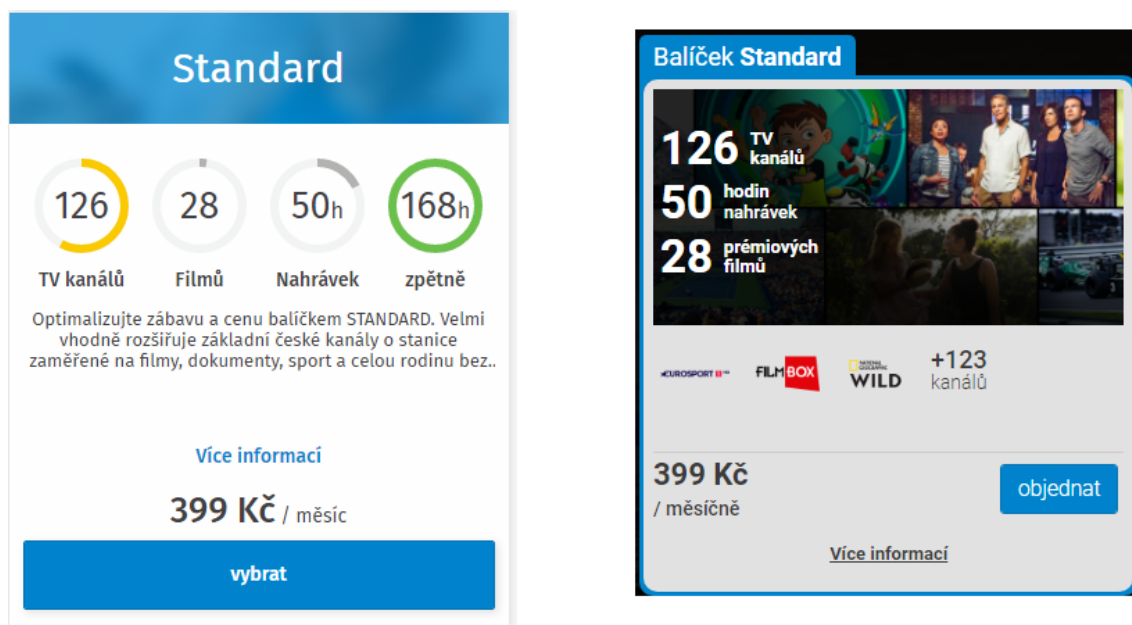
více stránek oddělených čárkami a to z toho důvodu, pokud by například některá stránka (v tomto kontextu reklama) nebyla přístupná pro konkrétního uživatele. Statické stránky se samy o sobě dají omezovat podle určitých pravidel. Je tedy například možné vytvořit stránku, která se má zobrazit pouze uživatelům, kteří jsou nově registrovaní (datum registrace není starší než dva týdny).

Poslední možností je pouze volba umístění hlavních balíčků. Opět se jedná pouze o to, že někteří partneři požadují mít hlavní balíčky napevno v horní části stránky. Kdežto výchozí nastavení zobrazuje tyto balíčky ve spodní části stránky, a to především pro uživatele, kteří již vlastní některý hlavní balíček, a tedy nejsou pro něj tolik relevantní.

## **Balíčky**

Detailní pojednání o službách společnosti a konkrétně o balíčcích, které seskupují některé tyto služby je více informací v kapitole [3.2.6](#). Dále je v této kapitole detailně vysvětlena logika, jak fungují specifikace, jak funguje položka ceníku a samotná výsledná služba. Vzhledem k tomu, že zaváděná úprava se týká webového obchodu, který prodává tyto balíčky, díky kterým lze následně sledovat pořady, je tato sekce jedna z nejdůležitějších. Současný vzhled balíčků lze přirovnat k některým webovým stránkám z minulého desetiletí. Oproti novému vzhledu je zde viditelný krok kupředu viz obrázek [4.13](#). Na levé straně je vidět starší verzi balíčků, která obsahuje čtyři kruhy, ve kterých je vyobrazen obsah balíčku. Vzhled starší verze byl staticky naprogramován, a tedy nebylo možné tyto balíčky nijak dynamicky přizpůsobovat.

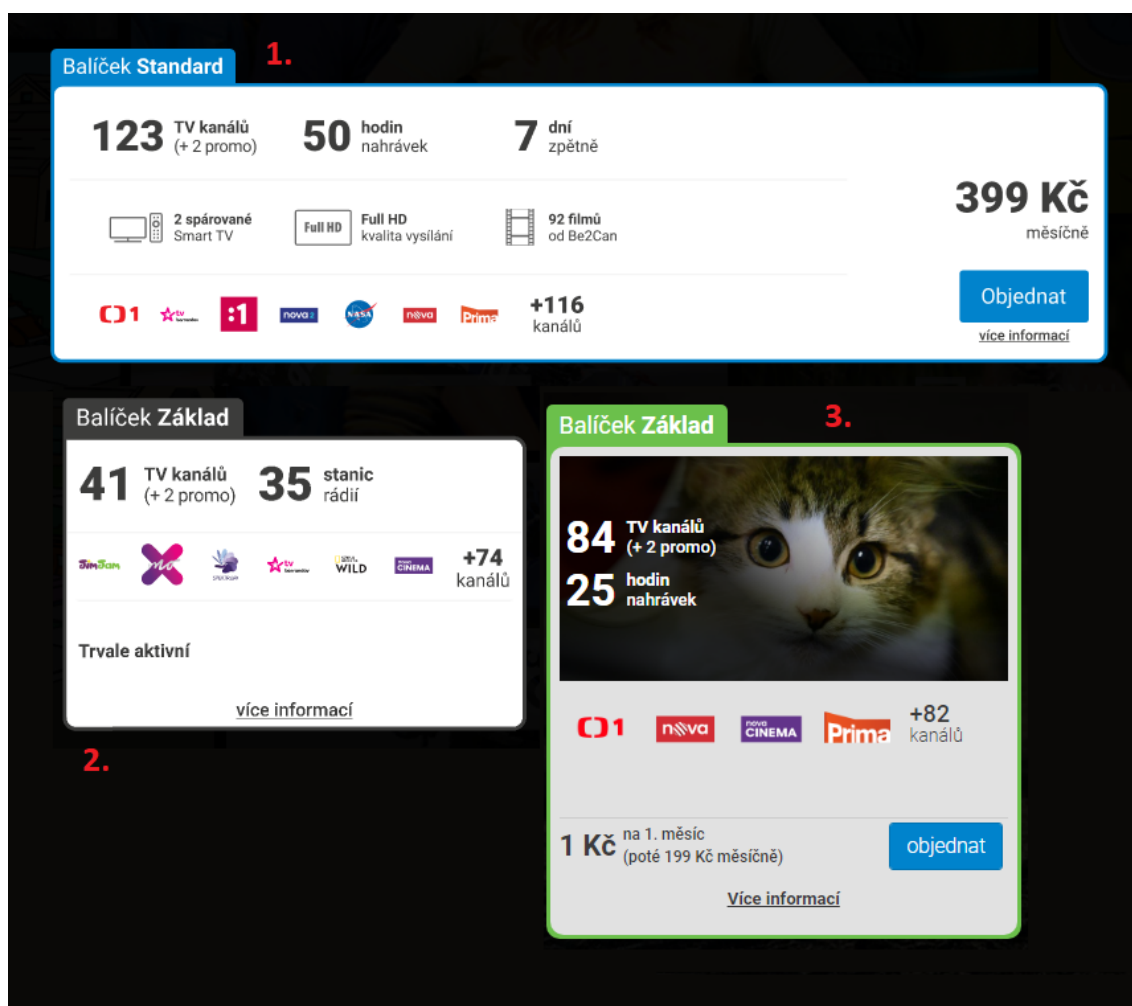




Obrázek 4.13: Porovnání balíčků (staré - levá strana, nové - pravá strana) (zdroj: vlastní tvorba)

Nová verze balíčků byla navržena v několika různých režimech. Tyto režimy se mimo samotného rozložení balíčků liší také velikostí. Existují přesně tři typy (velikosti) - malý, střední a velký. Pro lepší představu jsou návrhy těchto typů viditelné na obrázku 4.14. V horní polovině, označený číslem 1., je umístěn největší balíček (**large**), který by měl sloužit k vykreslování hlavních balíčků pro uživatele, kteří prozatím žádný balíček nemají, a tedy by tyto balíčky měly být nejvíce viditelné, není prozatím jisté, zdali tento design bude použitý, ale navržený a naprogramovaný bude. Navíc tento režim *velkého balíčku* bude využitý v některých jiných částech systému mimo samotného obchodu. Další režim velikosti balíčku je střední (**medium**), který je na obrázku označený číslem 3., tento režim bude sloužit pravděpodobně k vykreslováním hlavních balíčků pro uživatele, kteří již mají zakoupený některý balíček či případně pro zobrazování ostatních balíčků. Střední režim je zároveň také výchozí a bude pravděpodobně použit globálně na většinu balíčků. Poslední režim (číslo 2.) je nejmenší (**small**) velikost balíčku a v největší pravděpodobnosti bude využita pro minoritní balíčky, rozšiřující služby apod.

Mimo samotnou velikost balíčku je možné také detailní přizpůsobení balíčku, které bude lépe popsáno v následující kapitole. V jednoduchosti lze říci, že je možné změnit téměř cokoliv, co je viditelné v boxu balíčků, vyjímaje umístění tlačítka a ceny.



Obrázek 4.14: Balíčky - tři režimy zobrazení balíčku (zdroj: vlastní tvorba)

### Balíčky - přizpůsobení

Uživatelská část balíčků byla popsána v předchozí kapitole. Důležitá je taky část samotného přizpůsobení, které se bude vyskytovat v administraci. Každý partner/provozovatel/administrátor, jež bude mít specifická oprávnění, bude moci upravovat vzhled celého obchodu a jeho balíčků. Účel tedy je, aby například oddělení marketingu nebo obchodu, mohlo nastavovat veškeré informace o těchto balíčcích, především včetně vzhledu.

Na obrázku 4.15 je vidět základní nastavení balíčku. Jedná se o nastavení specifikace v záložce přizpůsobení ve velké administraci. Zde je možné nastavit základní vzhled balíčku, tak jak bude vypadat v elektronickém obchodě. Je zde několik důležitých nastavení, které je nutno zmínit. Základním prvkem je políčko **typ**, které může nabývat hodnot *Rich* nebo *Default*. Jedná se o přepínání starého a nového vzhledu balíčků. Je to zde umístěno z toho

důvodu, že může nastat situace, kdy některý partner či poskytovatel nechce využívat nový vzhled balíčků a chce zůstat u starého. Hodnota **rich** tedy zpřístupňuje všechno další nastavení, které je na obrázku vidět. Jedná se o nastavení barev pozadí hlavičky, orámování balíčku, barva textu v hlavičce. Následně lze nastavit i gradient pro samotné pozadí, aby byl například viditelný text na obrázkovém pozadí. Mimo vzhledových nastavení je zde také možnost zadat vlastní text, jež bude nastaven jako obsahu balíčku.

Jak již bylo dříve zmíněno, balíček je vlastně způsob, jakým společnost poskytuje služby svým zákazníkům. Tedy existuje několik druhů balíčků, například balíček poskytující možnost sledování několika desítek sportovních programů, či případně možnost sledovat pořady zpětně nebo stovky hodin nahrávek. Všechny tyto informace lze vykreslit do okna balíčku. Mimo jiné lze přímo nastavit i speciálně řádek kanálů, kde bude zobrazeno prvních několik kanálů, jež balíček zpřístupňuje. Uvnitř okna balíčku může být také vlastní text, který si zde administrátor nastaví, tento text se bude zobrazovat, pokud bude zároveň vyplněný a aktivovaný. Následně tento text nahradí vnitřek balíčku - lze použít například pro prodej různých speciálních služeb či produktů, které nelze nijak lépe vizuálně popsat.

V neposlední řadě jsou zde políčka pro skrývání či zobrazování různých dodatečných informací, jako skrytí slova *balíček* v názvu balíčků či případně skrývání časových hodnot, na jak dlouho je možné si balíček zakoupit.

Dále se zde nachází možnost *"Will be subset of specification"* tohle znamená, že například balíček **Základ** je pod-balíčkem balíčku **Standard**. Po nastavení této možnosti se například při prohlížení balíčku **Základ** uživateli zobrazí co vše by dostal navíc když by si zakoupil balíček **Standard** a může tak tedy lépe porovnat tyto dvě služby. Lze to samozřejmě porovnávat i naopak. Tato funkcionalita by měla klienty ponoukat k zakoupení lepších služeb, které jsou sice o něco dražší ale nabízí v přepočtu daleko větší množství výhod.

Customizace balíčku

Typ

Rich

Barva (pozadí hlavičky + orámování)

#0183cc

Barva text (hlavička)

☒ Gradient (pozadí)
 ☒ Zobrazit obrázek na pozadí

Row (channels)

Kanály

☐ Show special row (devices, quality, hbo, vod)
 ☒ Obsah - TV kanály
 ☐ Obsah - flexi kanály
 ☐ Obsah - rádia
 ☐ Obsah - TimeShift
 ☒ Obsah - nahrávky
 ☒ Obsah - VOD
 ☐ Content - HBO GO
 ☐ Content - HBO OD
 ☐ Zobrazit vlastní text

Obsah - vlastní text

Aplikovat pro (pls)

Standard (ID: 13057) (399 Kč)

Standard (ID: 21626) (399 Kč)

☐ Hide package word
 ☐ Hide package duration

Will be subset of specification

Náhled balíčku

Default - normal (fallback) no other changes are allowed, Rich - new custom design

Main color for background of package header + border

Secondary color - only text color in header

Enable/Disable gradient on background image (background image is required)

Show background image if there is any uploaded and design type is RICH

Content of special row (example - promo channels)

Show/hide special row (with promo channels)

Show in package - count of TV channels

Show in package - count of Flexi channels (if there are any)

Show in package - count of radio channels (if there are any)

Show in package - amount of timeshift (if there is any)

Show in package - amount of records (if there is any)

Show in package - count of VOD (if there are any)

Show in package - HBO GO

Show in package - HBO OD

Enable custom text (custom text needs to be filled)

If custom text is filled and enabled it will appear inside of package box instead of content above

By default keep it empty, if there is any item in pricelist which is sharing same ServiceSpecification you can specify for which one will be applied this design

Hide word package in headline of package

Hide package duration in shop and detail

Uložit

Obrázek 4.15: Přizpůsobení balíčku - základní nastavení (zdroj: vlastní tvorba)

85

## Editovat specifikaci služby Standard

[Základní nastavení](#)

Customizace balíčku

[Partner plugin](#)

Package detail

☒ Show platforms

☒ Show radios

☒ Icons - channel count

☒ Icons - timeshift

☒ Icons - HBOGO

☒ Icons - VOD

☒ Icons - records

☒ Icons - devices

☒ Icons - quality (fullHD/4K)

☒ Icons - Android/Apple

Static page connected to content

(žádný)

☒ Show bar (container) with platforms

☒ Show RADIO section - only if package has any radios

☒ Show box/icon how many channels has this package

☒ Show box/icon how many timeshift has this package allowed

☒ Show icon HBOGO but only if this packages has allowed HBOGO

☒ Show box/icon how many movies is in this packages

☒ Show box/icon how many free minutes/hours/days has this packages for recordings

☒ Show box/icon if this packages has allowed devices like SmartTV or Android

☒ Show box/icon which quality has this package (4K/FullHD)

☒ Show box/icon Android/Apple support

☒ Static page will be connected after last order button in detail

Obrázek 4.16: Přizpůsobení balíčku - nastavení detailu (zdroj: vlastní tvorba)

Další možnost přizpůsobení se týká především detailu balíčku viz obrázek 4.16. Tento detail je viditelný, pokud si uživatel obchodu klikne na možnost "Více informací" ve spodní části balíčku. Následně se mu zobrazí modální okno s obsahem balíčku. Opět je zde možnost skrývat/zobrazovat různé informace o službách, které balíček poskytuje.

Navíc je zde možnost připojit statickou stránku (viz kapitola 3.2.5), která bude vykreslena v obsahu tohoto detailu.

Copy package customization

Upřesnění

(žádný)

Kopírovat

Copy package customization from selected Specification into this one.

Náhled

Balíček Standard

124 TV kanálů (+ 3 promo)

50 hodin nahrávek

70 prémiových filmů

KINOSPORT

FILMBOX

+125 kanálů

123 Kč / měsíčně

objednat

Obrázek

Soubor: /services/background/1051.png

Pust soubor k přepsání

nebo

Vyber soubor

nebo

vlož URL

Nahraj z URL

Obrázek 4.17: Přizpůsobení balíčku - nastavení pozadí (zdroj: vlastní tvorba)

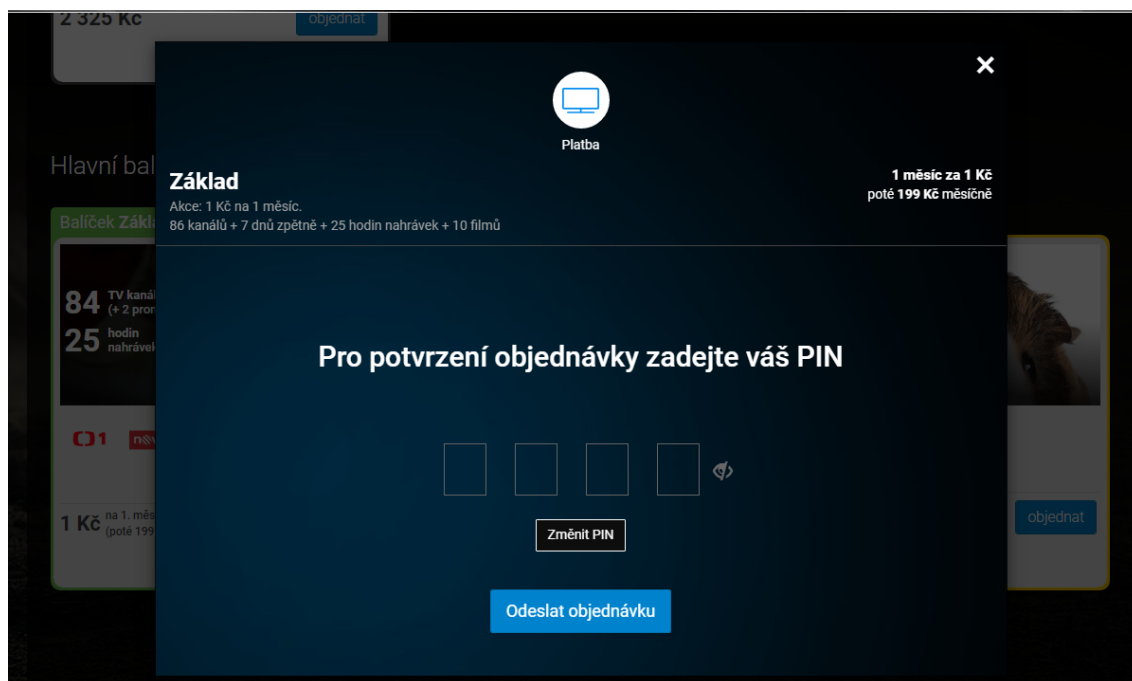
Poslední a opět docela důležitou částí přizpůsobení je nahrávání samotného obrázku pozadí. Náhled tohoto přizpůsobení je viditelný na obrázku 4.17. Po nahrání obrázku je samozřejmě nutné v přizpůsobení balíčku zatrhnout možnost "Zobrazit obrázek pozadí". Také je zde možnost zkopírovat celý vzhled z jiné specifikace, která je již nastavena. Sekce náhled (také viditelná na obrázku 4.17) soužší k zobrazení toho, jak balíček v současné době vypadá, aniž by administrátor musel uložit provedené změny. Tato sekce se zde nachází z toho důvodu, že všechny provedené změny se hned po uložení projeví na stránce obchodu. Je tedy nutné si dávat pozor na to, co se mění a pro testování vzhledu používat tlačítko "Náhled balíčku".

## Rychlá platba

Dalším z důvodů inovace elektronického obchodu, mimo samotný nový vzhled, je takové nový platební proces. Starý (respektive současný) obchod fungoval stejně jako většina elektronických obchodů. Klient si přidal zboží, přešel do košíku. Následně si zvolil platbu, například pomocí platební karty. Byl přesměrován na příslušnou platební bránu konkrétní společnosti (ČSOB, GoPay), kde musel zadat svoji platební kartu včetně všech potřebných údajů a provedl platbu na platební bráně. Následně byl přesměrován zpět na stránky obchodu.

Tento proces je poněkud zdlouhavý a mnoho uživatelů odradí od dokončení nákupu. Bylo tedy nutné tento proces upravit a zkrátit. Pro zkrácení tohoto procesu bude využit rodičovský zámek, někde pojmenován také jako uživatelský PIN. Tento zámek je čtyřmístný kód, který si klient může vytvořit či změnit ve své uživatelské sekci. Mimo jiné je tento zámek v současnosti využíván pro odemykání kanálu, jež jsou přístupné pouze osobám starších 18 let.

V případě, že je nový internetový obchod funkční, dokončený a aktivní. Je tedy možné využít zrychlený platební proces. Pokud uživatel, již někdy v minulosti platil pomocí platební karty, dostal možnost si tuto platební kartu uložit. V případě, že uživatel má tedy uloženou platební kartu, je mu tedy umožněno celý platební proces přeskočit pomocí rodičovského kódu a provést platbu během jednoho kliknutí. Je zde několik důležitých podmínek, které musí být splněny. Uživatel tedy musí mít zapamatovanou kartu pro konkrétní platební systém, tedy například pro **GoPay** či **ČSOB**. Balíček, který si chce uživatel zakoupit, respektive ceník, ve kterém balíček je, musí zároveň podporovat tento platební systém. Pokud jsou tyto klíčové podmínky splněny, tak při stisknutí tlačítka "Objednat" bude uživatel vyzván k zadání rodičovského kódu a následně bude provedena okamžitá platba viz obrázek 4.18.



Obrázek 4.18: Platební proces (zdroj: vlastní tvorba)

## Platební systém implementace

Implementace platebních systémů byla provedena již dříve a zde bude pouze stručně popsána. Pro lepší pochopení této sekce je nutné mít základní znalosti jazyka PHP, viz kapitola 2.3.4. Logika platebních systémů je poněkud náročná. Existuje zde jedna abstraktní třída *PaymentHandler*, která uchovává základní logiku zpracování plateb a komunikaci s platební bránou. Jelikož zde existuje několik druhů platebních systémů jako GoPay, PayU, CSOB apod. tak je bohužel nutné, aby každý systém měl naprogramován svoji vlastní logiku pro komunikaci s bránou pomocí API. Proto je zde vytvořena také řada tříd jako *GoPayPaymentHandler*, *PayUPaymentHandler* apod. které se starají o konkrétní komunikování s platební bránou pomocí API. Odesílají se zde různé unikátní klíče, které dokáží identifikovat příjemce platby, následně je zde možné odesílat uživatelskou kartu, pokud ji má zapamatovanou. apod. Všechny výše zmíněné třídy jsou volány a využity pomocí metod ve třídě *OrderApi*. Tato třída obaluje veškeré chování objednávek a celého objednávkového procesu. Vzhledem k tomu, že mimo webové aplikace se zde do budoucna počítá také s placením v samotných aplikacích, je nutné mít jednotné API aby, logika placení byla identická jak pro aplikace tak pro webový obchod.



Jak již bylo řečeno výše, třída *OrderApi* obsahuje řadu metod, které se starají o celý proces nákupu. Vkládání zboží do košíku, změna platebních systémů a jejich jednotlivých možností (zapamatovat platební kartu nebo automaticky prodlužovat). Následně jsou zde metody, které vyvolají finální platební proces. Tato metoda se nazývá *payOrder()*. Po zavolání této metody je dynamicky zvolen *PaymentHandler*, který se postará o zbytek platebního procesu. Specifický *PaymentHandler* je volen pomocí platebního systému přiřazeného k objednavce. Pokud ve chvíli, kdy je položka přidávána do košíku, neexistuje žádná aktivní objednávka, je vytvořena nová. Díky tomu, jaká položka se v košíku objeví, tak lze detekovat výchozí platební systém, pomocí kterého se bude platba provádět. V novém platebním modelu (v novém obchodě) je využit pouze pomyslný košík (pouze na pozadí). Uživatel si bude moci zakupovat pouze jednu položku současně. Toto rozhodnutí vycházelo ze statistik o nákupu více položek najednou. Existovalo nepatrné množství uživatelů (2% - 3%), kteří při nákupním procesu zakoupili více než jednu věc z obchodu. Vzhledem k tomu, že se zde jedná o celkové zjednodušení nákupního procesu, tak mezikrok, kde uživatel musel jít do košíku, byl úplně vynechán.

Výběr platebních systémů včetně volby různých speciálních možností, jako je například **zapamatování karty**, jež bylo zmíněno dříve, je znázorněn v následujícím obrázku 4.19.

**Způsob platby**

☒  
  
**Platební kartou**

**Přes GoPay**

Objednávka bude zpracována okamžitě.

**Automaticky prodlužovat balíčky**  
 Balíčky budou před jejich vypršením automaticky prodlouženy.

**Zapamatovat platební kartu**  
 Pro další nákupy.

☒  
  
☒

Odeslat objednávku

\* povinná pole

Obrázek 4.19: Platební proces - výběr platebního systému (zdroj: vlastní tvorba)

### Proces objednávky

Jak již bylo popsáno výše v kapitole 4.7.2, objednávka začíná stisknutím tlačítka *objednat*, jež se nachází na každém balíčku v internetovém obchodě. Stisknutí tohoto tlačítka vyvolá modální okno a zároveň na pozadí volá metodu *addItemToCart()* z třídy *OrderApi*. Metoda přidá položku (balíček) do pomyslného košíku a následně dynamicky vykreslí obsah modálního okna. Obsahem okna může být jednak rychlá platba (kapitola 4.7.2) nebo výběr platebního systému (obrázek 4.19). Třída *OrderApi* v tuto chvíli již uchovává aktivní objednávku, která je uložena v databázi v tabulce ***Order*** (viz návrh relační databáze v kapitole 4.10). Každá akce, která je následně provedena v tomto modálním okně ovlivňuje přímo záznam objednávky v databázi (změna platebního systému, různé nastavení objednávky, opakované prodlužování apod.).

Ve chvíli kdy uživatel stiskne tlačítko *Odeslat objednávku* je ve třídě *OrderApi* zavolána metoda *payOrder()*, která podle zvoleného platebního systému (vycházející z abstraktní třídy *PaymentHandler*) zavolá metodu *pay()*.

Každý platební systém má implementovanou vlastní logiku placení, která komunikuje s platební bránou. Metoda *pay()* v platebním systému vytvoří transakci, která slouží jako úložiště pro veškeré informace odeslané a přijaté z platební brány. Po vytvoření transakce je odeslána žádost na bránu. Zároveň je vytvořen záznam v tabulce *OrderService*, který propojuje objednávku a službu, jež bude následně aktivována. V případě, že je platba úspěšně provedena, je zavolána metoda *confirmOrder()*. Tato metoda provede potvrzení objednávky, označí stav jako zaplacená a potvrzená. Následně vytvoří požadovaný počet služeb do tabulky *Service* a nastaví jim časový interval ve kterém budou tyto služby aktivní.

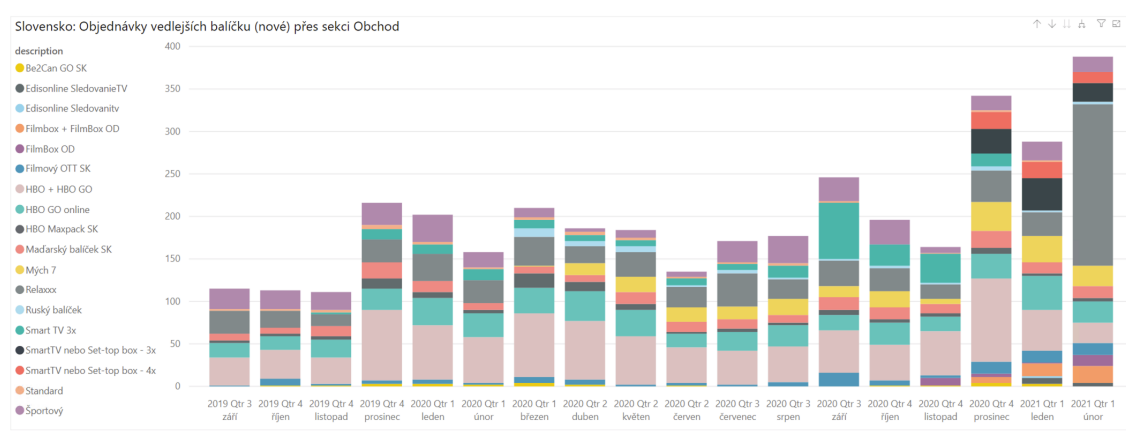
Tento proces musel být na několika místech upravený. Vzhledem k tomu, že předchozí nákupní proces nepodporoval platby pomocí rodičovského zámku, tak bylo nutné tento PIN dostat až do třídy *OrderApi*. Zde se sám nový obchodní model rozhoduje, zdali využije platbu pomocí PINu nebo nikoliv. Jak již bylo dříve řečeno, jednou z podmínek pro využití možnosti potvrzení objednávky pomocí rodičovského pinu byla uložena platební karta uživatele pro konkrétní platební systém (například GoPay/ČSOB). Následně se zde ještě rozhoduje, zdali se jedná o nákup balíčku obsahující pouze služby nebo například produkt. Produkty zde mají jednu menší výjimku a tou je nutnost zadání doručovací adresy.

Mimo podmínky, aby měl klient uloženou platební kartu u konkrétního platebního systému, zde existuje ještě jedno pravidlo. Klient by měl mít nastavený svůj rodičovský kód. V případě, že ho nemá, je mu nabídnuta možnost tento kód vytvořit během platebního procesu.

## 4.8 Přínosy

V případě, že nasazení nového projektového řízení, konkrétně agilní metodiky SCRUM, bylo úspěšné, byl realizovatelný vývoj nového webového obchodu pro společnost. Vývoj projektů v týmu je bez vhodného projektového řízení velice neefektivní a leckdy nerealizovatelný. To byl důvod, proč zde popisovaná společnost musela využít nové agilní projektové řízení.

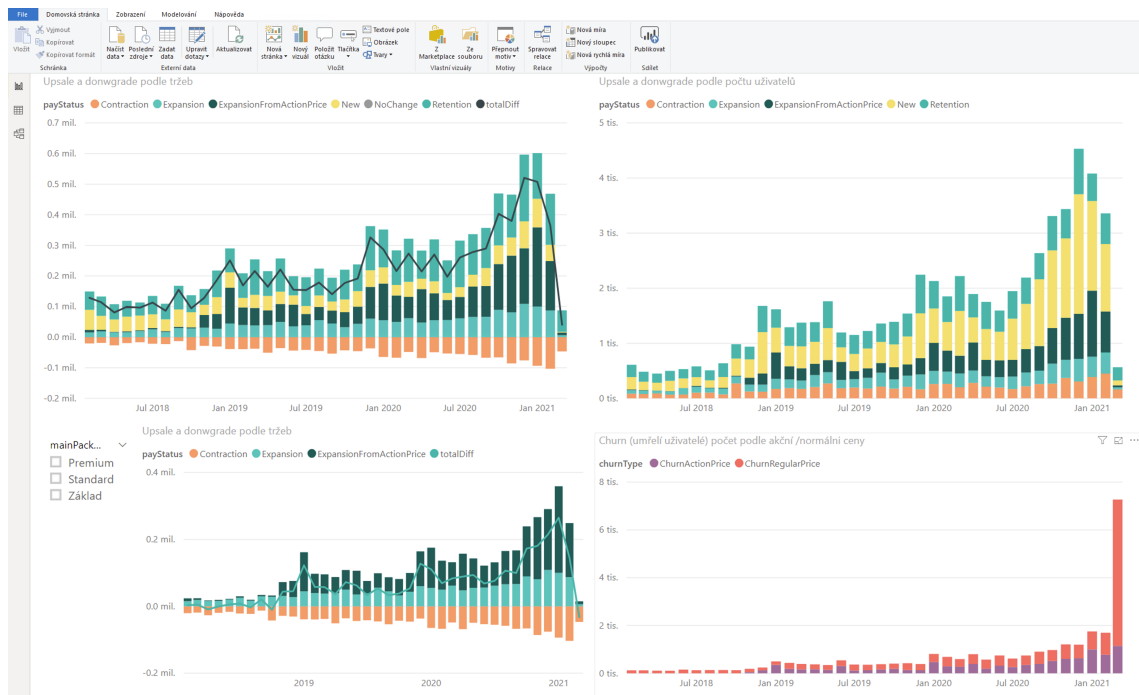
Za pomoci toho řízení byl navržen a implementován webový obchod. Tento nový obchod má jednodušší a moderní vzhled, který se snaží usnadňovat a zpříjemňovat zákazníkům jejich nákup. Na prvním grafu 4.20 lze vidět znatelný nárůst klientů, kteří si zakoupili balíčky a to počínaje **prosincem** roku **2020**, kdy byl nasazen tento nový webový obchod.



Obrázek 4.20: Graf objednávek balíčků (zdroj: marketing SledovaniTV s.r.o.)

Další obrázek 4.21 zobrazuje čtyři různé grafy. První graf zobrazuje tržby, které generují zákazníci, kteří si různě modifikovali (zlepšovali nebo snižovali) svoje služby. Další graf na stejném obrázku popisuje opět stávající uživatele, jež si modifikovali své služby (upgrade/downgrade), ale zohledňuje zde počty těchto uživatelů. Třetí graf (levý spodní) zobrazuje opět tržby zákazníků, ale jsou zde zohledněny pouze tři hlavní balíčky (**Základ**, **Standard**, **Premium**). Poslední graf zobrazuje počty tzv. odumřelých zákazníků.

Z předchozích grafů lze tedy s jistotou říci, že po nasazení nového webového obchodu (koncem roku 2020) vzrostly tržby společnosti, a tedy i celkový počet stávajících zákazníků, kteří se prozatím společnost drží a dále využívají její služby. Vzhledem k tomu, že tento nový obchodní model cílí z velké části na své stávající zákazníky, byly tyto průzkumy prováděny na upgradech/downgradech balíčků především u stávajících zákazníků.



Obrázek 4.21: Graf platících zákazníků a jejich balíčků (zdroj: marketing SledovaniTV s.r.o.)

# Závěr

Tato diplomová práce byla zaměřena primárně na návrh a implementaci dílčí části informačního systému za využití nového agilního projektového řízení. Konkrétně se jednalo o návrh a implementaci nového webového obchodu. Celý tento návrh a implementace byla popisována na společnosti poskytující IPTV. To znamená, že společnost poskytuje televizní vysílání za pomoci internetového připojení. Zde popisovaná společnost již dříve vlastnila internetový obchod, ve kterém nabízela svoje služby prostřednictvím tzv. balíčků. Obsah těchto balíčků zpřístupňoval určité televizní kanály, filmy či různé další služby.

První část diplomové práce popisuje **teoretická východiska** a různé technologie použité při návrhu a implementaci dílčí části informačního systému. Dále jsou zde popsány různé principy projektového řízení a konkrétně informace o agilní metodice SCRUM.

Další částí práce je **analýza problému a současné situace**. Součástí této kapitoly, je mimo představení společnosti a její struktury, také řada analýz. Byly zde využity analýzy pro zkoumání interních faktorů společnosti, či případně pojednání o vnějších vlivech na společnost. Výsledkem těchto analýz byla řada důvodů, kvůli kterým se společnost mohla rozhodnout inovovat určité části svého informačního systému. Důvody byly především tyto: **aktuální konkurence na trhu a vysoká vyjednávací síla klientů**. Další výsledky analýz určily také nedostatečné projektové řízení ve společnosti. Bez nového a efektivního projektového řízení by byl vývoj jakýchkoliv části systému finančně a časově velice náročný.

Poslední část práce je věnována **návrhu a implementaci řešení** a následnému **vyhodnocení přínosů**. Návrh je i zde rozdělen do dvou částí. První částí je integrace nového agilního projektového řízení - SCRUM do společnosti. Jsou zde popsány také některé nástroje, které budou využity pro zefektivnění tohoto řízení. Druhou částí této kapitoly je návrh a implementace nového webového obchodu. Důvodem inovace, respektive vývojem nového webového obchodu je především silná stávající konkurence, která se neustále

zvětšuje a nabízí svým zákazníkům modernější a spolehlivější možnosti nákupu služeb. Webový obchod, jež by klientovi dovolil velice rychle a jednoduše zakoupit oblíbený balíček televizních kanálů, by tedy společnosti velice prospěl a dokázal by ji prozatím udržet konkurenceschopnou. Je zde také časový a finanční plán pro návrh a implementaci těchto dvou částí do společnosti a jejího systému.

Já jakožto autor této práce, jsem ve zde zmiňované společnosti, působil jako programátor, který vyvíjel tento internetový obchod a zároveň byl přítomen od začátku integrace nového projektového řízení.

# Literatura

- [1] ADMOSPHERE, N. *Vývoj sledovanosti TV*. [Online; navštíveno 13.4.2020]. Dostupné z: <https://www.nielsen-admosphere.cz/press/ato-koronavirova-krize-potvrdila-dominanci-televize-mezi-medii/>.
- [2] ALBRIGHT, D. *What Is IPTV? Everything You Need to Know About the Future of Video* [online]. [cit. 2021-15-01]. Dostupné z: <https://www.uscreen.tv/blog/what-is-iptv/>.
- [3] ATLISSIAN. *Jira*. [Online; navštíveno 13.4.2020]. Dostupné z: <https://www.atlassian.com/software/jira>.
- [4] BOURDON, R. *WAMP*. [Online; navštíveno 9.4.2020]. Dostupné z: <https://www.wampserver.com/en/>.
- [5] BOURGEOIS, D. T. *Information Systems for Business and Beyond*. Saylor Foundation, 2014. ISBN 9781304943484.
- [6] CEKATELEVIZE.CZ. *Česká televize - informace*. [Online; navštíveno 13.4.2020]. Dostupné z: <https://www.ceskatelevize.cz/vse-o-ct/zakladni-informace-o-ct/>.
- [7] CONTRIBUTORS, A. G. *Apache.org*. [Online; navštíveno 9.4.2020]. Dostupné z: [https://httpd.apache.org/ABOUT\\_APACHE.html](https://httpd.apache.org/ABOUT_APACHE.html).
- [8] FINANCE.CZ. *Mobilní operátoři v ČR*. [Online; navštíveno 13.4.2020]. Dostupné z: <https://www.finance.cz/518730-mobilni-operatori-v-cesku/>.
- [9] FOCUSAGE.CZ. *Rekordní sledovanost televize*. [Online; navštíveno 13.4.2020]. Dostupné z: [https://www.focus-age.cz/m-journal/aktuality/rekordni-brezen--sledovanost-televize-byla-vlivem-pandemie-koronaviru-nejvyssi-za-17-let\\_\\_s288x15152.html](https://www.focus-age.cz/m-journal/aktuality/rekordni-brezen--sledovanost-televize-byla-vlivem-pandemie-koronaviru-nejvyssi-za-17-let__s288x15152.html).
- [10] GDPR. *GDPR*. [Online; navštíveno 13.4.2020]. Dostupné z: <https://www.gdpr.cz/>.
- [11] GRUDL, D. *Dibi*. [Online; navštíveno 9.4.2020]. Dostupné z: <https://dibiphp.com/cs/>.
- [12] ING.TOMÁŠ HAJÍČEK, M. *Porterova analýza*. [Online; navštíveno 13.4.2020]. Dostupné z: <https://vseomarketingu.estranky.cz/clanky/marketing/porterova-analyza.html>.



- [13] INTERNETPROVSECHNY.CZ. *Internetové připojení - rozšířenost*. [Online; navštíveno 13.4.2020]. Dostupné z: <http://www.internetprovsechny.cz/podle-ctu-je-nejrozsirenejsim-pripojenim-k-internetu-pres-wifi-roste-optika/>.
- [14] (ITNETOWKR.CZ), D. Čápka. *MVC*. [Online; navštíveno 9.4.2020]. Dostupné z: <https://www.itnetwork.cz/php/mvc/objektovy-mvc-redakcni-system-v-php-popis-architektury>.
- [15] JQUERY.COM. *jQuery*. [Online; navštíveno 9.4.2020]. Dostupné z: <https://jquery.com/>.
- [16] KLČOVÁ, H. a SODOMKA, P. *Informační systémy v podnikové praxi*. Computer Press, 2011. ISBN 978-80-251-2878-7.
- [17] KUNCE, E. a ŠOCHOVÁ, Z. *Agilní metody řízení projektů*. Computer Press, 2014. ISBN 978-80-251-4194-6.
- [18] LAURENČÍK, M. *SQL - Podrobný průvodce uživatele*. Grada, 2018. ISBN 978-80-271-0774-2.
- [19] LAURENČÍK, M. *Tvorba www stránek v HTML a CSS*. Grada, 2019. ISBN 978-80-271-2241-7.
- [20] LIBBY, A. *Instant LESS CSS Preprocessor How-to*. Packt Publishing Limited, 2013. ISBN 9781782163763.
- [21] MACH, O. *Co je to projektové řízení?* [Online; navštíveno 9.4.2020]. Dostupné z: <https://www.shean.cz/clanky/detail/co-je-to-projektove-rizeni.htm>.
- [22] MANAGEMENTMANIA.COM. *Co je Lewinův třífázový model změn (Lewin's Three-Stage Model of Change)*. [Online; navštíveno 13.4.2020]. Dostupné z: <https://managementmania.com/cs/lewinuv-trifazovy-model-zmen>.
- [23] MANAGEMENTMANIA.COM. *Co je McKinsey 7S?* [Online; navštíveno 13.4.2020]. Dostupné z: <https://managementmania.com/cs/mckinsey-7s>.
- [24] MANAGEMENTMANIA.COM. *Co je PESTLE analýza?* [Online; navštíveno 13.4.2020]. Dostupné z: <https://managementmania.com/cs/pestle-analyza>.
- [25] MANAGEMENTMANIA.COM. *Řízení projektů (Project Management)*. [Online; navštíveno 9.4.2020]. Dostupné z: <https://managementmania.com/cs/metody-rizeni-projektu>.
- [26] MARIADB.COM. *MariaDB*. [Online; navštíveno 9.4.2020]. Dostupné z: <https://mariadb.org/about/>.
- [27] MYSLÍN, J. *Elektronická kniha: Scrum*. Computer press, 2016. ISBN

- 978-80-251-4650-7.
- [28] MYSQL.COM. *MySQL*. [Online; navštíveno 9.4.2020]. Dostupné z:  
<https://www.mysql.com/about/>.
- [29] NETTE.ORG. *jQuery*. [Online; navštíveno 9.4.2020]. Dostupné z:  
<https://nette.org/cs/>.
- [30] PODNIKATEL.CZ. *Zákon o obchodních společnostech*. [Online; navštíveno 13.4.2020].  
Dostupné z: <https://www.podnikatel.cz/zakony/zakon-o-obchodnich-spolecnostech-a-druzstvech-zakon-o-obchodnich-korporacich/uplne/>.
- [31] POŘÍZEK, J. *SWOT analýza a její využití*. [Online; navštíveno 13.4.2020]. Dostupné z:  
<https://www.ecommercebridge.cz/swot-analyza-a-jeji-vyuziti/>.
- [32] PRETTYMAN, S. *Learn PHP 7: object oriented modular programming using HTML5, CSS3, Javascript, XML, JSON, and MYSQL*. Apress, 2015. ISBN 978-1-4842-1730-6.
- [33] SCM.COM git. *GIT*. [Online; navštíveno 9.4.2020]. Dostupné z:  
<https://git-scm.com/book/cs/v2/%C3%9Avod-Z%C3%A1klady-syst%C3%A9mu-Git>.
- [34] VYSOKESKOLY.CZ. *Hodnocení vysokých škol v ČR*. [Online; navštíveno 13.4.2020].  
Dostupné z: <https://www.vysokeskoly.cz/tagy/hodnoceni-vysokych-skol>.
- [35] ČNB. *Dopad pandemie COVID-19 na ekonomiku ČR*. [Online; navštíveno 13.4.2020].  
Dostupné z: [https://www.cnb.cz/cs/o\\_cnb/cnblog/Prvni-odhad-dopadu-pandemie-COVID-19-na-ekonomiku-CR/](https://www.cnb.cz/cs/o_cnb/cnblog/Prvni-odhad-dopadu-pandemie-COVID-19-na-ekonomiku-CR/).
- [36] ČSÚ. *Vývoj obyvatelstva v ČR*. [Online; navštíveno 13.4.2020]. Dostupné z:  
<https://www.czso.cz/documents/10180/121739318/13015720.pdf/91360556-5d5a-48bb-995a-8b40faeee7e0?version=1.1>.
- [37] ŽÁRA, O. *JavaScript - Programátorské techniky a webové technologie*. Computer Pres, 2015. ISBN 9788025145739.

# Seznam obrázků

2.1	MVC - diagram (zdroj itnetwork.cz) [14]	18
2.2	Příklad návrhu relační databáze (zdroj: vlastní tvorba)	19
3.1	Graf vývoje sledovanosti TV podle technologií (zdroj viz [1])	27
3.2	Struktura společnosti (zdroj: vlastní tvorba)	28
3.3	Uživatelská sekce (zdroj: vlastní tvorba)	31
3.4	Administrace systému (zdroj: vlastní tvorba)	32
3.5	Partnerská administrace (zdroj: vlastní tvorba)	33
3.6	Ukázka customizace (zdroj: vlastní tvorba)	34
3.7	Ukázka současného internetového obchodu (zdroj: vlastní tvorba)	37
4.1	Graf časové analýzy (zdroj: vlastní tvorba)	58
4.2	Jira - ukázka sprintu (zdroj: vlastní tvorba)	65
4.3	Jira - ukázka úkolu (zdroj: vlastní tvorba)	67
4.4	GitLab - ukázka projektu (zdroj: vlastní tvorba)	68
4.5	GitLab - ukázka seznamu requestů	69
4.6	GitLab - ukázka seznamu requestů (zdroj: vlastní tvorba)	70
4.7	Grafický návrh webového obchodu (zdroj: vlastní tvorba)	73
4.8	Mapa rizik (zdroj: vlastní tvorba)	75
4.9	Mapa rizik po opatření (zdroj: vlastní tvorba)	77
4.10	ERD diagram objednávek (zdroj: vlastní tvorba)	78
4.11	Jira sprint - Epic webového obchodu (zdroj: vlastní tvorba)	79
4.12	Přízpůsobení obchodu (zdroj: vlastní tvorba)	80
4.13	Porovnání balíčků (staré - levá strana, nové - pravá strana) (zdroj: vlastní tvorba)	82
4.14	Balíčky - tři režimy zobrazení balíčku (zdroj: vlastní tvorba)	83

4.15	Přizpůsobení balíčku - základní nastavení (zdroj: vlastní tvorba)	85
4.16	Přizpůsobení balíčku - nastavení detailu (zdroj: vlastní tvorba)	86
4.17	Přizpůsobení balíčku - nastavení pozadí (zdroj: vlastní tvorba)	87
4.18	Platební proces (zdroj: vlastní tvorba)	89
4.19	Platební proces - výběr platebního systému (zdroj: vlastní tvorba)	91
4.20	Graf objednávek balíčků (zdroj: marketing SledovaniTV s.r.o.)	93
4.21	Graf platících zákazníků a jejich balíčků (zdroj: marketing SledovaniTV s.r.o.)	94

# Seznam tabulek

3.1 Swot analýza . . . . .	47
4.1 Síly působící na změnu . . . . .	54
4.2 Náklady na technologie/nástroje . . . . .	59
4.3 Souhrn nákladů . . . . .	61
4.4 Tabulka rizika jejich ohodnocení (zdroj: vlastní tvorba) . . . . .	74
4.5 Rozdělení mapy rizik (zdroj: vlastní tvorba) . . . . .	75
4.6 Tabulka rizik - opatření (zdroj: vlastní tvorba) . . . . .	76